

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky

DIPLOMOVÁ PRÁCE

2011

Bc. Tomáš Piprek

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra informatiky

Rozšíření IS pro Základní Školu
U Kapličky 959, Orlová

Extension of IS for Elementary School
U Kapličky 959, Orlová

Zadání diplomové práce

Student:

Bc. Tomáš Piprek

Studijní program:

N2647 Informační a komunikační technologie

Studijní obor:

2612T025 Informatika a výpočetní technika

Téma:

Rozšíření IS pro Základní Školu U Kapličky 959, Orlová
Extension of IS for Elementary School U Kapličky 959, Orlová

Zásady pro vypracování:

Diplomová práce navazuje na bakalářskou práci a dále ji rozšiřuje. Student má za úkol zhodnotit použití IS od nasazení do školy před 2 lety a rozšířit funkčnost stávajícího IS o další moduly - třídní kniha, evidence známek, jejich přehledy a statistiky, vytváření testů a testování, další tiskové sestavy dle požadavků.

1. Proved'te analýzu provozu IS od doby spuštění IS v rámci bakalářské práce a popište případné změny, které IS způsobil v provozu školy.
2. Zjistěte požadavky na moduly "třídní kniha" tak, aby umožňoval stejnou nebo lepší funkčnost jako současná "papírová" třídní kniha (probírané učivo, absence), dále na modul "vytváření testů a dotazníků" pro uživatele (testy z předmětů, evaluační dotazníky) a dále na modul pro získání přehledů známek dle požadavků učitelů, vypracujte analýzu a implementujte ke stávajícímu IS.
3. Navrhněte a implementujte změny pro přizpůsobení IS pro on-line provoz ve škole (v každé třídě je už dostupný počítač).
4. Navrhněte a implementujte spolupráci IS s ostatními programy používanými ve škole (importy, exporty).
5. Vzhledem k plánovanému zrušení papírových dokladů navrhněte potřebné tiskové výstupy z IS a implementujte.
6. Podle požadavků uživatelů a analýzy současného provozu řešte případné další úpravy IS.
7. Zjistěte vhodnost Nette framework pro implementaci modulů, popište možnosti tohoto frameworku a technologie AJAX, DRY, KISS, MVC a výše vyjmenované moduly implementujte pomocí tohoto frameworku.
8. Navrhněte a implementujte vhodné data miningové metody pro analýzu výše uvedených testů, dotazníků a výsledků žáků.

Seznam doporučené odborné literatury:

Podle pokynů vedoucího diplomové práce.

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

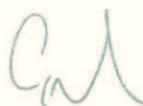
Vedoucí diplomové práce: **Ing. Emilie Šeptáková**

Datum zadání: 20.11.2009

Datum odevzdání: 06.05.2011



doc. Dr. Ing. Eduard Sojka
vedoucí katedry



prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prohlášení studenta

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Orlové 4. května 2011

.....

Poděkování

Chtěl bych poděkovat vedoucí diplomové práce Ing. Emilii Šeptákové za vstřícný přístup a pomoc při tvorbě této diplomové práce a dále ředitelům škol ZŠ U Kapličky Mgr. Kamilu Paloncymu, ZŠ Karla Dvořáčka Mgr. Zdeňku Nowakovi a ICT metodikům Bc. Martinu Czajovi a Mgr. Jiřině Koplové za ochotu a trpělivost při tvorbě této diplomové práce a implementaci IS.

Abstrakt

Tato diplomová práce navazuje na bakalářskou práci – IS pro Základní Školu U Kapličky 959, Orlová. Po testovacím provozu a ostrém spuštění IS požadovala škola další rozšíření funkcí. Například o třídní knihu nebo různé statistiky a výpisy známek. Během provozu IS na Základní Škole U Kapličky projevila o IS zájem také Základní Škola Karla Dvořáčka 1230 v Orlové. Diplomová práce se zabývá požadavky těchto dvou škol, jejich implementací a také popisem a zkušenostmi s provozem IS na těchto školách. Implementace nových funkcí IS bude provedena pomocí frameworku NETTE a vzhledem k tomu, že se jedná o nový framework a navíc od českého autora, bude se část diplomové práce věnovat také bližšímu popisu tohoto frameworku. V závěrečné části diplomové práce bude také uvedena analýza dat získaných při provozu IS pomocí dataminingových metod.

Klíčová slova:

informační systém, žákovská knížka, datamining, třídní kniha, statistiky, funkční analýza, datová analýza, PHP, MySQL, NETTE, ZŠ U Kapličky 959, Orlová – Lutyně, ZŠ Karla Dvořáčka 1230, Orlová – Lutyně, zkušenosti, www.zsukaplicky.cz, www.zsdvoracka.cz

Abstract

This diploma thesis follows on bachelor's thesis - IS pro Základní Školu U Kapličky 959, Orlová. After testing phase and deployment of IS, the school requested extension of functionalities. For example the classes register, statistics and mark reports. During operation of IS on elementary school U Kapličky, the elementary school Karla Dvořáčka 1230 in Orlová also requested the implementation of this IS.

Diploma thesis deals with requests of both schools, their implementation and with description of experiences with operation of IS. Implementation of new functionalities will be done in NETTE framework. And because of this framework is new and comes from Czech developer, part of the thesis will be dealing with this new framework.

In the final phase of thesis, there will be analysis of data stored during operation of IS using datamining methods.

Keywords:

administrator, teacher, parent, pupil, student, information system, report book, functional analysis, data analysis, PHP, MySQL, elementary school, ZŠ U Kapličky 959, Orlová – Lutyně, e-learning, www.zsukaplicky.cz

Seznam použitých zkratk a symbolů

AJAX	Asynchronous JavaScript and XML
BI	Business intelligence
BIDS	Business Intelligence Developement Studio
CSRF	Cross site request forgery
CSS	Cascading style sheets
CSV	Coma separated values
DB	Databáze
DS	Datový sklad
DRY	Don't repeat yourself
ETL	Extract, transform, load
FORPSI	Poskytovatel webhostingu, www.forpsi.com , INTERNET CZ, a.s.
HTML	HyperText Markup Language
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol over Secure Socket Layer
HW	Hardware
IS	Informační systém
IT	Informační technologie
KISS	Keep it short and simple
MVC	Model – view – controller
MVP	Model – view – presenter
OLAP	Online analytical processing
OLTP	Online transaction processing
OS	Operační systém
PHP	Personal Home Page
SQL	Structured query language
SW	Software
WWW	World Wide Web
XHTML	Extensible HyperText Markup Language
XSS	Cross site scripting
ŽK	Žákovská knížka

Obsah

1	Úvod	3
1.1	Cíle diplomové práce	3
1.2	Situace na základních školách obecně.....	3
1.3	Situace na Základní škole U Kapličky 959, Orlová-Lutyně.....	4
1.3.1	WWW stránky.....	4
1.3.2	Spuštění jednotných emailových schránek.....	4
1.3.3	Zkušenosti s provozem IS	5
1.4	Situace na Základní škole Karla Dvořáčka, Orlová	5
1.5	Požadavky škol.....	6
2	Zadání.....	7
2.1	Podrobná specifikace zadání	8
2.2	Funkční požadavky	10
2.2.1	Vstupy	10
2.2.2	Výstupy	10
2.2.3	Role uživatelů.....	11
2.2.4	Okolí.....	12
2.3	Nefunkční požadavky.....	13
3	Datová analýza.....	14
3.1	Lineární zápis	14
3.2	Popis vztahů:	16
3.3	ERD.....	17
3.4	Datový slovník	19
4	Funkční analýza	21
4.1	Kontextový diagram.....	21
4.2	DFD 0. úrovně.....	22
4.3	DFD 1. úrovně - Funkce pro učitele.....	23
4.4	Minispecifikace	24
4.4.1	Vytvoření výpisu známek.....	24
4.4.2	Vložení záznamu do třídní knihy	25
4.4.3	Vytvoření testu.....	26
4.4.4	Zobrazení výpisu známek.....	27
4.4.5	Hromadné generování uživatelů.....	28
5	Návrh implementace.....	29
5.1	Upřesnění požadovaných SW a HW technologií	29
5.2	Popis jednotného konfiguračního souboru	29
5.3	Výkon	29
5.4	Zálohování.....	30
5.5	Uživatelské rozhraní.....	30
6	Implementace.....	32
6.1	HW/SW	32
6.2	Popis frameworku NETTE.....	32
6.3	Stručný popis postupu při vytváření aplikací v NETTE frameworku	37
6.3.1	Instalace frameworku	37

6.3.2	Struktura aplikace.....	38
6.3.3	Presenter.....	38
6.3.4	Šablona.....	40
6.3.5	Model.....	42
6.4	Popis protokolu HTTPS.....	44
6.4.1	Popis protokolu HTTP.....	44
6.4.2	Popis protokolu HTTPS.....	45
6.4.3	Certifikáty a certifikační autority.....	45
6.4.4	Shrnutí provozu protokolu HTTPS.....	46
6.5	Instalace.....	47
6.6	Popis jednotného konfiguračního souboru.....	47
6.7	Použití HTTPS protokolu.....	47
6.8	Popis implementace nového typu uživatele – sekretářka.....	48
6.9	Popis implementace importů žáků.....	50
6.10	Popis implementace generování PDF výpisů známek.....	50
6.10.1	Technické řešení implementace.....	51
6.10.2	Ukázka struktury výpisu známek.....	51
7	Testování.....	52
7.1	Testovací data.....	52
7.2	Zaškolení.....	54
8	Analýza dat.....	55
8.1	Popis datových skladů.....	55
8.1.1	Dimenze a fakta.....	55
8.1.2	Multidimenzionální model.....	56
8.1.3	Datový sklad.....	56
8.1.4	Tvorba datového skladu.....	56
8.1.5	ETL.....	56
8.2	Popis vlastního datového skladu.....	57
8.2.1	Popis zdrojových dat.....	57
8.2.2	Popis zdrojových tabulek a atributů.....	57
8.2.3	Rozdělení na dimenze a fakta.....	58
8.2.4	Schéma datového skladu.....	59
8.2.5	Nepoužité atributy.....	60
8.2.6	Popis programového vybavení.....	60
8.2.7	Integrace dat - ETL.....	60
8.2.8	Přehled statistických údajů.....	62
8.2.9	Vytvoření OLAP kostky.....	63
8.3	OLAP analýza a reporty.....	64
8.3.1	Počty známek podle předmětů.....	65
8.3.2	Průměrné známky podle předmětů.....	65
8.3.3	Počty známek podle měsíců.....	66
8.3.4	Průměry známek podle tříd.....	67
8.3.5	Průměry známek podle učitelů.....	68
8.4	Popis dataminingu.....	69
8.4.1	Popis programového vybavení pro datamining.....	69

8.4.2	Příprava dat pro datamining	69
8.4.3	Rozhodovací stromy	70
8.4.4	Shlukovací metody	71
8.4.5	Asociace	71
8.5	Zhodnocení analýzy dat	72
9	Závěr.....	73
9.1	Základní škola U Kapličky.....	73
9.1.1	Zhodnocení provozu IS	73
9.1.2	Ostatní IT.....	73
9.2	Základní škola Karla Dvořáka.....	74
9.2.1	Zhodnocení provozu IS	74
9.2.2	Analýza dat.....	74
9.3	Do budoucna	75

1 Úvod

V rámci bakalářských prací IS pro Základní školu U Kapličky 959, Orlová [1] a IS pro Základní školu v Lošticích [2] byl spuštěn na výše uvedených základních školách informační systém (dále jen IS), ze kterého vychází tato diplomová práce. Od doby spuštění IS na Základní škole U Kapličky 959 v Orlové (dále jen ZŠ U Kapličky) už uběhly zhruba dva roky. Toto je pro provoz informačního systému relativně dlouhá doba. Během této doby se uskutečnila celá řada změn. Osobně hodnotím velmi kladně pokrok školy za toto období. Konkrétní zkušenosti budou blíže popsány v kapitole Zkušenosti s provozem IS. Během provozu se pochopitelně objevily některé provozní problémy s IS a také byla objevena spousta možností pro vylepšení. Postupně vznikaly různé požadavky na další funkce IS. O rok později byl IS spuštěn spolu s novými www stránkami i na Základní škole Karla Dvořáčka 1230 v Orlové (dále jen ZŠ Karla Dvořáčka). Vzhledem k tomu, že se jednalo již o třetí školu, na které byl IS spuštěn (detaily o provozu na Základní škole v Lošticích jsou popsány v kapitole Testování, zaškolení a nasazení do provozu v [2]), bylo zde spuštění mnohem jednodušší a úpravy existujícího IS už nebyly tak náročné.

Během používání IS však školy zjistily, že by využily i další funkce jako například rozšířené výpisy a přehledy známek, elektronickou evidenci třídní knihy nebo rozhraní pro online testování žáků.

1.1 Cíle diplomové práce

Cílem diplomové práce bude popsat zkušenosti se současným provozem IS na ZŠ U Kapličky a ZŠ Karla Dvořáčka a navrhnout a implementovat požadovaná rozšíření a změny, které vznikly na základě požadavků jednotlivých škol. Při tvorbě dalších funkcí bude kladen důraz na dodržování obecného návrhu IS, který se osvědčil v rámci bakalářské práce, a který umožnil jednoduché použití na dalších školách a přidávání nových funkcí.

V závěru bude uvedena ukázka použití dataminingových metod pro analýzu dat získaných při provozu IS.

1.2 Situace na základních školách obecně

Vzhledem ke zkušenostem z prostředí základních škol budou nejprve na úvod shrnuty obecné zkušenosti.

Dle mého názoru se za dobu vývoje IS, což jsou zhruba poslední dva roky, situace na základních školách postupně zlepšovala. Do jisté míry to je způsobeno postupným rozšiřováním IT technologií mezi čím dále tím širší laickou veřejností. Od dřívějšího strachu a neochoty pracovat s výpočetní technikou lidé, dle mých zkušeností, přecházejí k názoru, že jim výpočetní technika a obecně IT technologie začínají znatelně usnadňovat život. Toto se projevuje i ve školství. Je třeba si však uvědomit, že situace na základních školách je stále na výrazně horší úrovni než na středních nebo vysokých školách. Tato postupná změna přístupu k IT technologiím se velmi kladně projevuje v běžném životě a zejména v prostředí základních škol. I u samotných zaměstnanců je možné

pozorovat zájem o nové technologie a hlavně snaha učit se nové věci. Kromě snahy a zájmu uživatelů bylo možné také pozorovat zájem rodičů.

Vzhledem k tomu, že Orlová je relativně malé město, ve kterém je 8 základních škol, rodiče z jiných škol se brzy dozvěděli o provozu IS na ZŠ U Kapličky. To jen zvýšilo zájem rodičů o spuštění IS na ZŠ Karla Dvořáčka. Zde jde jasně vidět pokrok ve vnímání IT, kdy jej běžní lidé začínají brát jako standardní součást života a aktivně se o tyto technologie zajímají.

Podporu pro zavádění informačních technologií lze také vidět i na straně jednotlivých obcí, krajů i Evropské unie. Během uplynulých dvou let byly školy v Orlové vybaveny veškerou potřebnou IT infrastrukturou v rámci projektu „Dejte nám šanci poznat svět“. Každá škola nyní disponuje připojením k internetu v každé místnosti, dataprojektorem a interaktivní tabulí. V rámci softwarové výbavy byly pořízeny interaktivní učebnice nakladatelství FRAUS a další elektronické výukové materiály.

1.3 Situace na Základní škole U Kapličky 959, Orlová-Lutyně

1.3.1 WWW stránky

Jak již bylo zmíněno, od spuštění IS na ZŠ U Kapličky uplynuly téměř dva roky. Na začátku škola neměla ani funkční www stránky ani vlastní doménu (jak je popsáno v kapitole Situace na Základní škole U Kapličky 959, Orlová-Lutyně [1]). Osobně jsem velice spokojen s vývojem situace na této škole. Po počátečním náročném zpracování textů, kdy se v podstatě začínalo od úplného začátku, byly úspěšně spuštěny www stránky s jednoduchým redakčním systémem, a hlavně se povedlo zajistit pro www stránky trvalý přísun aktuálních informací. Na stránkách jsou stovky fotografií z mnoha školních akcí, různá videa a další zajímavý obsah. Kvalitní a aktuální informace vedly ke stále a poměrně vysoké návštěvnosti www stránek.

V rámci www stránek také funguje schránka důvěry, prostřednictvím které už několik žáků kontaktovalo speciálního pedagoga se svými problémy.

Protože je správa www stránek vzhledem ke svému rozsahu značně náročná, byly určeny osoby z řad zaměstnanců školy, které si rozdělily povinnosti týkající se www stránek. Hlavní osobou je ICT metodik školy, který provádí téměř všechny změny na www stránkách. Z učitelů českého jazyka byla vybrána osoba zodpovědná za jazykovou korekturu obsahu stránek. Významnou sekci týkající se školního poradenského pracoviště má na starosti speciální pedagog, který také řeší dotazy odeslané prostřednictvím schránky důvěry. Důležité změny jsou také konzultovány s ředitelem a vedením školy.

Ukázky z obsahu stránek a statistiky přístupů jsou uvedeny v příloze A.

1.3.2 Spuštění jednotných emailových schránek

Zavádění IS do jakékoli organizace není jen o jednotlivých technických řešeních. Důležité je také definovat role jednotlivých uživatelů a prostředky pro jejich komunikaci. Současná situace, kdy zaměstnanci školy měli na www stránkách školy uvedeny své soukromé emailové adresy v různých tvarech a na různých bezplatných e-mailových serverech, se ukázala jako nepřipustná.

Proto bylo po spuštění www prezentace rozhodnuto vytvořit všem zaměstnancům školy jednotné emailové adresy ve tvaru `jmeno.prijmeni@zsukaplicky.cz`. Tímto je zpřehledněna a zkvalitněna komunikace mezi zaměstnanci školy a žáky a rodiči. Pro technické řešení byla vybrána služba Google Apps [10]. Tato služba poskytuje bezplatné a spolehlivé řešení pro provoz e-mailových adres na vlastní doméně. Kromě technického řešení byla také stanovena pravidla pro používání nově vytvořených emailových schránek, aby se například nestalo, že zaměstnanci tyto schránky nebudou pravidelně kontrolovat. Bližší popis je však nad rámec této diplomové práce.

1.3.3 Zkušenosti s provozem IS

Velmi mě potěšily reakce rodičů, kteří se již před spuštěním www stránek a hlavně před spuštěním samotného IS často ptali, kdy už budou moci využívat tyto služby. Spuštění IS podporovalo také vedení školy, což je v kombinaci se zájmem samotných uživatelů dle mého názoru ideální situace pro spuštění a provoz IS.

Na druhou stranu zadávání dat do IS přináší pochopitelně práci navíc pro všechny zaměstnance školy. Vzhledem k tomu, že bylo zavádění IS podporováno jak rodiči, tak vedením školy, jsem v průběhu provozu nenarazil na žádnou neochotu a žádné stížnosti zaměstnanců školy ohledně nadbytečné práce ve spojitosti s novým IS.

Při počátečním provozu se pochopitelně objevovala řada menších problémů s provozem IS a jednotlivými funkcemi. Zde se velmi osvědčila spolupráce s ICT metodiky školy – s paní Mgr. Jiřinou Koplovou ze ZŠ U Kapličky a panem Bc. Martinem Czajou ze ZŠ Karla Dvořáčka, kteří se starali o provoz IS a průběžně řešili běžné problémy s provozem. Díky tomu stačilo všechny problémy řešit s těmito ICT metodiky, kteří průběžně shromažďovali požadavky jednotlivých uživatelů a zasílali informace o chybách. Tento způsob značně zjednodušil a zpřehlednil komunikaci s uživateli IS vzhledem k tomu, že reálných uživatelů jsou řádově stovky.

Uvedení ICT metodici se také starali o rutinní provoz IS. Toto bylo důležité například pro obyčejné generování zapomenutých hesel. Často se stávalo, že rodiče psali na můj email uvedený v IS pro případné řešení technických problémů a žádali o zaslání hesla. Toto pochopitelně není z bezpečnostních důvodů možné a v praxi se toto řešilo osobním předáváním hesel prostřednictvím ICT metodiků.

Celkově lze vnímat snahu o rozšíření informačních technologií a jejich každodenní běžné používání. Zde lze pozorovat změnu oproti původní situaci, kdy se používaly pouze programy pro evidenci údajů žáků, jak bylo vyžadováno příslušnými zákony, a jakékoli další možnosti nad tento nutný rámec nebyly používány.

1.4 Situace na Základní škole Karla Dvořáčka, Orlová

1.4.1 WWW stránky

Před spuštěním samotného IS byla i pro tuto školu vytvořena www prezentace s jednoduchým redakčním systémem obdobně jako na ZŠ U Kapličky. Předchozí www stránky byly značně nevyhovující a vzhledem k částečné provázanosti www stránek s IS bylo vytvoření www stránek

školy následované zavedením IS logickým krokem. Popis této prezentace je však nad rámec této diplomové práce. Detaily o www prezentaci školy, jako jsou ukázky obsahu stránek nebo statistiky přístupů, jsou uvedeny v příloze A.

1.4.2 Zkušenosti s provozem IS

Na ZŠ Karla Dvořáčka se spuštění IS pro známkování žáků a další evidenci potřebných záležitostí uvažovalo zhruba dva roky. Za tuto dobu se však nepovedlo IS spustit vzhledem k problémům s původním plánovaným dodavatelem. Po zkušenostech s provozem na ZŠ U Kapličky bylo rozhodnuto o spuštění stejného IS. Žákovská knížka tedy byla spuštěna na ZŠ Karla Dvořáčka o rok později než na ZŠ U Kapličky, tj. na začátku školního roku 2009/2010.

Spuštění IS na dvou školách nebylo možné zvládnout v rámci bakalářské práce. Během tohoto roku se jak vedení školy, tak i někteří rodiče a žáci dozvěděli o spuštění IS na ZŠ U Kapličky. Tato zvědavost rodičů spolu s ročním provozem IS, který tak byl již zbaven většiny počátečních problémů, umožnila výrazně jednodušší spuštění IS.

1.5 Požadavky škol

Jak již bylo zmíněno v bakalářské práci, po analýze požadavků škol bylo zjištěno, že tyto požadavky jsou až na některé detaily stejné. Toto se potvrdilo i v případě ZŠ Karla Dvořáčka, která byla již třetí školou, na které byl spuštěn IS.

Tato skutečnost byla zohledňována při analýze požadavků jednotlivých škol a značně tak usnadnila jejich přípravu. V praxi stačilo nové požadavky řešit s jednou školou a téměř ve všech případech šlo dané řešení použít na ostatních školách.

Konkrétní požadavky budou specifikovány v následující kapitole.

2 Zadání

Požadavky pro specifikaci zadání vznikly na základě skutečného provozu IS. Toto umožnilo, že byly postupně implementovány funkce IS na základě priorit. Zde se opět osvědčil postup, kdy uživatelé (zaměstnanci, rodiče a žáci) komunikovali průběžně s ICT metodikem školy, se kterým poté byly na pravidelných schůzkách společně s vedením školy konzultovány jednotlivé požadavky, které byly následně implementovány.

Požadavky byly evidovány už od prvního spuštění IS. Vzhledem k delší době, kdy byly evidovány, jich bylo poměrně hodně. Detailně budou popsány dále.

Při spouštění IS do ostrého provozu a při hromadném vytváření uživatelských účtů se ukázalo, že tyto rutinní záležitosti jsou značně náročné. Je tedy třeba implementovat příslušné funkce pro hromadné vkládání dat. V některých případech, jako je například generování přístupových údajů, je třeba také vytvořit hromadné operace pro výstupy dat.

Protože se do budoucna plánuje celkové zrušení papírové evidence známek a další agendy, což jsou žákovské knížky a třídní knihy, je třeba připravit funkce pro zhotovení výpisů a vyřešit jejich předávání uživatelům, zejména pak žákům.

Během provozu se také ukázalo, že vzhledem k běžným administrativním úkonům je potřeba často získávat z IS různá data. Tyto záležitosti mají na starosti pracovníci sekretariátu školy, pro které je třeba zřídit nový typ uživatele se zvláštními přístupovými právy.

Známkování žáků však pokrývá jen omezenou část školní agendy. Další významnou částí je evidence třídní knihy. Po zkušenostech s prvotním provozem IS a postupném připravování technického zázemí bylo rozhodnuto o realizaci dalšího modulu IS pro třídní knihu, který bude evidovat zejména probíranou výuku a absence žáků.

Základní školy mají ze zákona povinnost provádět v určitých intervalech různá dotazníková šetření (autoevaluační dotazníky) a navíc pravidelně zjišťují například spokojenost žáků a rodičů s výukou a další podobné dotazníkové průzkumy. Zde tedy vznikl další požadavek na realizaci takovýchto testů a dotazníků v elektronické podobě.

Během doby, kdy byl IS v provozu, se nahromadilo značné množství dat, jako například známky žáků, rozvrhy a podobně. V současné době však školy běžně vyhodnocují data jen základními statistickými metodami. Proto tedy zaměstnance uvedených škol a zejména jejich vedení zajímá možnost dalšího využití těchto dat. Nashromážděná data tedy budou analyzována s použitím různých dataminingových metod. Vzhledem k tomu, že tato oblast je pro zaměstnance základních škol novým tématem, bude se tato analýza dat zabývat touto problematikou od úplného počátku.

2.1 Podrobná specifikace zadání

Následující popis bude přibližně strukturován na základě časové posloupnosti vývoje jednotlivých funkcí. Jako první budou uvedeny funkce, které školy požadovaly během prvotního provozu IS a dále nově vyvíjené funkce v rámci této diplomové práce.

- **Hromadné operace**
- **Sekretářka**
- **Výpisy známek**
- **Přehledy známek a statistiky**
- **Třídní kniha**
- **Dotazníky a testy**
- **Datamining**

- **Hromadné operace**

Při ostrém spouštění IS se ukázalo značně náročné počáteční vkládání dat uživatelů. I přesto, že administrační rozhraní bylo vytvořeno s důrazem na co nejsnazší obsluhu a byly již implementovány některé funkce pro hromadné úpravy, jako například generování žáků a jejich rodičů podle celé třídy, vkládání stovek nových uživatelů bylo i přesto náročné. Bude tedy třeba navrhnout a implementovat funkce pro administrátora s cílem ulehčit rutinní zadávání počátečních dat. Bude se jednat o importy dat, které ze souboru ve formátu xls, případně csv načtou data žáků, které se získají z programu DM Evidence, který používá většina základních škol pro evidenci osobních údajů žáků dle odpovídajících zákonů. Administrátor tedy exportuje data z výše uvedeného programu a případně doplní chybějící údaje, které nebudou v exportovaném souboru dostupné, jako například loginy žáků. Tato data se poté importují do předem vytvořených tříd.

Dalším logickým krokem je generování přístupových údajů pro vytvořené uživatele. Vzhledem k tomu, že již byla stanovena pravidla pro formáty loginů rodičů, lze vygenerovat v jednom kroku jak přístupové údaje pro žáky, tak i pro jejich příslušné rodiče. Administrátor po importu žáků může zvolit generování přístupových údajů, kde se na základě jmen a loginů žáků hromadně vygenerují hesla pro všechny žáky a loginy a hesla pro jejich rodiče. Zde lze ušetřit značnou práci s vkládáním osobních údajů rodičů tím, že rodiče budou muset tyto údaje vyplnit sami při prvním přihlášení do IS, což jednak zajistí aktuálnost vložených informací a také značně ušetří práci administrátorům IS.

- **Sekretářka**

Jak již bylo výše uvedeno, na základě požadavků škol bylo rozhodnuto o vytvoření dalšího typu uživatele. Při návrhu IS v rámci bakalářské práce nebylo zřejmé, že bude třeba tento typ uživatele. V běžném provozu se však ukázalo, že kromě běžné práce s IS, kterou provádějí již existující typy uživatelů, je třeba také často získávat různé přehledy a výpisy dat uložených v IS. Je tedy třeba přidat typ uživatele, který se bude jmenovat sekretářka a který bude mít k dispozici v podstatě všechny výpisy dat, které jsou v IS. Tento typ uživatele však nebude moci žádná data editovat ani mazat, což zaručí bezpečnost IS.

- **Výpisy známek**

Z důvodu již zmiňovaného plánovaného zrušení papírových žákovských knížek je třeba připravit funkci, která bude zajišťovat tisk týdenních nebo měsíčních výpisů známek pro dané žáky. Výpisy musí sloužit jako plná náhrada žákovské knížky. Musí tedy obsahovat základní osobní údaje a dále přehled známek ve stejném formátu jako je v žákovské knížce a na který jsou jak učitelé, tak i žáci a jejich rodiče zvyklí. Vzhledem k množství žáků je třeba, aby byly výpisy sestaveny s ohledem na co nejmenší náročnost na počet stránek, které bude potřeba vytisknout.

- **Přehledy známek a statistiky**

Pro učitele je velmi užitečné, když mohou srovnávat celkové výsledky tříd a když mají k dispozici různé statistické údaje o výsledcích žáků, které mohou dále porovnávat a analyzovat. Je proto třeba sestavit přehledy známek za zvolené třídy, které budou obsahovat souhrnné výsledky za třídu a statistické údaje jako například průměr, medián, minima a maxima známek a podobně. Zde je třeba opět zohlednit současné používané statistiky a případně podle potřeby navrhnout další výsledky statistik. Tyto statistiky budou plnit funkci pro analýzu známek v reálném čase. Statistiky pololetních výsledků žáků jsou již řešeny programem DM Evidence a ukázka těchto statistik je uvedena v příloze F.

- **Třídní kniha**

Třídní kniha bude jedním ze tří základních modulů IS. Moduly pro známkování a správu rozvrhů již byly implementovány. Třídní kniha bude navržena s cílem zjednodušit, zrychlit a zpřehlednit rutinní správu školní agendy. Návrh třídní knihy by měl zohlednit to, že údaje budou zadávány přímo v hodině, což je umožněno dostupností počítače v každé třídě. Učitel tedy na začátku hodiny vyplní příslušný popis probíraného učiva a ze seznamu vybere chybějící žáky. Těmto chybějícím žákům se automaticky vygeneruje zpráva o absenci, kterou uvidí i rodiče. Pro evidovanou absenci a probírané učivo budou k dispozici i příslušné výpisy pro kompetentní uživatele.

- **Dotazníky a testy**

Častou náplní výuky jsou pochopitelně různé testy žáků. Několikrát do roka je také potřeba oslovit různými dotazníky i rodiče nebo zaměstnance školy. Z tohoto důvodu je potřeba implementovat rozhraní pro vytváření, vyplňování a vyhodnocování testů a dotazníků. Učitel může vytvořit vlastní test, kde budou definovány příslušné otázky a odpovědi. Dále pak nastaví, pro koho má být daný test přístupný. Vybraným uživatelům se poté zobrazí test, který mohou vyplnit a výsledky se poté zobrazí tvůrci testu.

- **Datamining**

Ve školách se často sestavují různé přehledy a statistiky. Nejčastěji se týkají známek jednotlivých žáků. Po diskuzi s vedením obou škol bylo rozhodnuto, že v rámci této diplomové práce bude navrženo řešení, které by umožňovalo jednak statistické analýzy výsledků žáků a také

pokročilé analýzy pomocí dataminingových metod. Cílem je, aby vedení školy získalo přehled o možnostech takovéto analýzy dat a v praxi mohlo vyzkoušet jednotlivé výstupy a výsledky analýzy. Očekávanými výstupy by měly být informace sloužící k porovnávání výsledků žáků podle tříd, učitelů a období školních roků a hledání různých závislostí v těchto datech.

2.2 Funkční požadavky

Požadavky:

- Umožnění importu žáků,
- hromadné generování přístupových údajů pro žáky a jejich rodiče,
- zpřístupnění údajů z IS pro sekretariát školy,
- rozhraní pro tisk přehledů známek,
- statistické přehledy výsledků žáků dostupné učitelům,
- evidence probírané výuky,
- evidence absence žáků,
- automatické generování výpisů o absenci pro žáky a rodiče,
- rozhraní pro vytváření testů a dotazníků,
- rozhraní pro vyhodnocování testů a dotazníků.

2.2.1 Vstupy

- Údaje žáků potřebné pro jejich import do IS,
- údaje žáků pro vygenerování jejich hesel a jejich rodičů,
- formulář pro definici výpisu známek,
- formulář pro definici výpisu třídní knihy,
- formulář pro nastavení parametrů pro přehledy známek,
- formulář pro zadávání probíraného učiva,
- formulář pro zadávání absence žáků,
- formulář pro vytvoření testu.

2.2.2 Výstupy

- Výpisy údajů pro novou uživatelskou roli – sekretářka,
 - seznam zaměstnanců,
 - seznam žáků,
 - seznam rodičů,
 - seznam tříd,
 - seznam místností,
 - seznam předmětů,
 - výpis rozvrhů,
 - výpis suplování,
- výpis vygenerovaných přístupových údajů pro žáky a jejich rodiče vytvořený funkcí pro generování údajů,
- výpis třídní knihy obsahující probírané učivo,

- výpis třídní knihy obsahující chybějící žáky,
- výpis známek žáků přizpůsobený pro tisk,
- výpis známek podle zvolených parametrů pro učitele,
- výpis absence pro žáka,
- výpis absence pro rodiče,
- výpis absence pro učitele,
- výpis probíraného učiva,
- test nebo dotazník pro rodiče nebo žáka,
- přehledy vytvořených testů a dotazníků pro učitele,
- výsledky testu nebo dotazníku pro učitele.

2.2.3 Role uživatelů

Během provozu IS vznikla potřeba vytvořit další typ uživatele, kterým je uživatel Sekretářka. Důvodem byla potřeba zpřístupnit přehledy o žácích, jejich známkách a absenci pro různé přehledy, které je potřeba evidovat a kontrolovat. Aby však zůstala zachována bezpečnost IS, bylo zakázáno pro takovéto úkoly vytvářet uživatele s právy administrátora, což by vedlo k výraznému snížení bezpečnosti IS. Proto byl vytvořen uživatel Sekretářka, který má přístup k vybraným údajům, avšak jen pro čtení.

Ostatní uživatelské role se osvědčily a nebylo třeba je měnit.

Administrátor (administrátor, ředitel, zástupce ředitele, ICT metodik):

- Importy žáků,
- generování žáků a jejich rodičů.

Učitel

- Rozhraní pro vytváření testů,
- rozhraní pro zobrazování výsledků testu,
- formulář pro zadávání probíraného učiva do třídní knihy,
- formulář pro zadávání absence,
- výpisy přehledů známek.

Rodič

- Zobrazení výpisu probíraného učiva podle údajů v třídní knize,
- zobrazení absence podle údajů v třídní knize,
- zobrazení dostupných testů,
- zobrazení konkrétního testu.

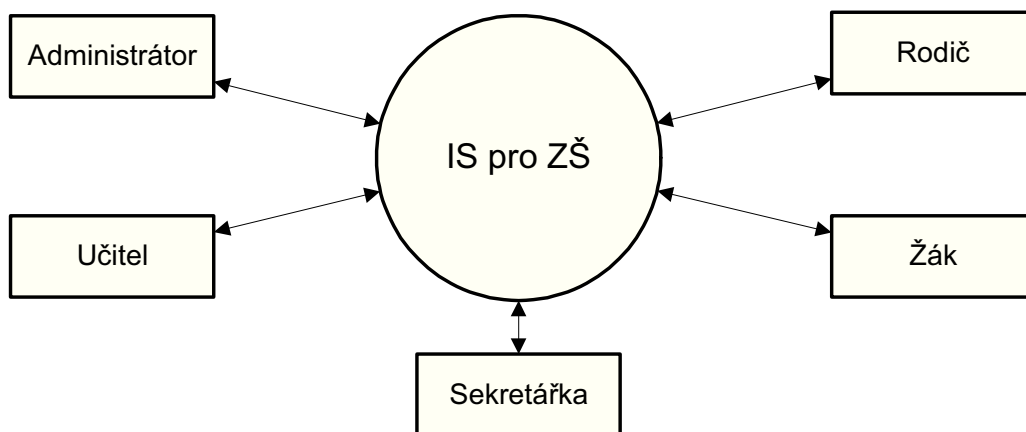
Žák

- Zobrazení výpisu probíraného učiva podle údajů v třídní knize,
- zobrazení absence podle údajů v třídní knize,
- zobrazení dostupných testů,
- zobrazení konkrétního testu.

Sekretářka

- Výpisy všech známek,
- výpisy žáků,
- výpisy učitelů,
- výpisy zaměstnanců,
- výpisy tříd,
- výpisy rozvrhů,
- výpisy předmětů,
- výpisy známek,
- výpisy z třídní knihy.

2.2.4 Okolí



Obrázek č. 1: Kontextový diagram

2.3 Nefunkční požadavky

V rámci bakalářské práce nebyly vyřešeny všechny možné způsoby zabezpečení. Vzhledem k tomu, že zpočátku nebyly v IS ukládány žádné citlivé údaje jako rodná čísla, nebylo toto třeba. Nyní se však počítá s evidencí všech údajů žáků. Je proto třeba připravit zbývající část zabezpečení, kterou je implementace komunikace mezi serverem a klientem prostřednictvím protokolu HTTPS.

Dále je třeba klást důraz na rychlost a jednoduchou použitelnost IS. Na tyto aspekty byl brán ohled již při tvorbě IS v rámci bakalářské práce. Nyní již není potřeba řešit tyto záležitosti od začátku. Je však třeba zhodnotit úspěšnost navrhnutého řešení s ohledem na uvedenou rychlost a použitelnost a navrhnout případné změny a další optimalizaci.

Vzhledem k tomu, že se základní školy potýkají často s nedostatkem financí, je také třeba optimalizovat finanční náklady na provoz a údržbu IS.

3 Datová analýza

Datové analýze byla věnována velká pozornost při návrhu IS v rámci bakalářské práce. Tento přístup se velmi vyplatil. Co se týče struktury databáze, nebylo třeba v průběhu provozu provádět žádné změny, kromě záležitostí pro nové funkce IS. Jen výjimečně bylo potřeba přidat, odebrat nebo upravit některý z atributů. Tyto atributy jsou označeny **tučně a podtržené**.

Nové tabulky, sloužící funkcím implementovaným v rámci této diplomové práce, jsou uvedeny na konci seznamu.

3.1 Lineární zápis

[primární klíč, *cizí klíč*]

Zak	(zak_id , zak_login, zak_heslo, zak_jmeno, zak_prijmeni, zak_ulice, zak_mesto, zak_psc, zak_e-mail, zak_datum_narozeni, zak_rodne_cislo, zak_okres_narozeni, zak_misto_narozeni, zak_narodnost, zak_statni_prislusnost, zak_pojistovna, zak_druzina, zak_jidelna, zak_teldomu, zak_mobil, zak_smazan, <u>zak_pohlavi</u> , <u>zak_absolvent</u>)
Zamestnanec	(zamestnanec_id, zamestnanec_login, zamestnanec_heslo, zamestnanec_titul_pred, zamestnanec_jmeno, zamestnanec_prijmeni, zamestnanec_titul_z, zamestnanec_kabinet, zamestnanec_email, zamestnanec_aprobace, zamestnanec_telefon, zamestnanec_role, zamestnanec_smazan)
Rodic	(rodic_id , rodic_login, rodic_heslo, rodic_titul_pred, rodic_jmeno, rodic_prijmeni, rodic_titul_z, rodic_ulice, rodic_mesto, rodic_psc, rodic_zamestnavatel, rodic_funkce, rodic_telefon_domu, rodic_telefon_prace, rodic_mobil, rodic_e-mail, rodic_smazan, rodic_vyplneno)
Rodic_zak	(rodic_zak_id , <i>rodic_id</i> , <i>zak_id</i>)
Aktualni_trida	(trida_id , trida_rocnik, trida_oznaceni, <i>mistnost_id</i> , <i>zamestnanec_id</i> , <i>rok_id</i> , trida_poznamka, trida_smazana)
Zak_trida	(zak_trida_id , <i>trida_id</i> , <i>zak_id</i> , zak_trida_smazano)
Mistnost	(mistnost_id , mistnost_zkratka, mistnost_nazev, mistnost_kapacita, mistnost_poznamka, mistnost_smazana)
Rozvrh	(rozvrh_id , <i>trida_id</i> , <i>mistnost_id</i> , <i>uci_id</i> , rozvrh_den, rozvrh_cislo_hodiny, rozvrh_smazan, <i>rok_id</i>)

Predmet	(predmet_id, predmet_zkratka, predmet_nazev, predmet_poznamka, predmet_smazan)
Zmena	(zmena_id, zmena_datum, rozvrh_id, zamestnanec_id, zmena_odpada, zmena_poznamka)
Uci	(uci_id, zamestnanec_id, predmet_id, uci_poznamka, uci_smazano)
Skolni_rok	(rok_id, rok_rok, rok_zacatek_pol, rok_konec_pol, rok_aktualni)
Vysledek	(vysledek_id, zak_id, predmet_id, vysledek_datum, vysledek_popis, vysledek_znamka, vysledek_vaha, zamestnanec_id, vysledek_precteno, rok_id, vysledek_smazano)
Poznamka	(poznamka_id, zak_id, rodic_id, poznamka_datum, poznamka_text, poznamka_precteno, zamestnanec_id, poznamka_precteno_datum, rok_id)
Sdeleni_rodic	(sdeleni_rodic_id, sdeleni_rodic_rodic_id, sdeleni_rodic_zamestnanec_id, sdeleni_rodic_datum, sdeleni_rodic_text, rok_id)
Elearning	(elearning_id, elearning_nadpis, elearning_text, elearning_cesta, uci_id, <u>elearning_datum, trida_id</u>)
Nové tabulky:	
Absence	(absence_id, zak_id, tridni_kniha_id, absence_popis)
Test	(test_id, test_nazev, test_popis, zamestnanec_id, test_vytvoril_zamestnanec_id, test_od, test_do, test_zak_id)
Test_odpovedi	(test_odpovedi_id, otazka_id, zak_id, test_odpoved)
Test_otazka	(test_otazka_id, test_otazka_text, test_otazka_poznamka, test_id)
Test_zak	(test_zak_id, zak_id, test_id)
Test_otazka_moznosti	(test_otazka_moznosti_id, test_otazka_moznosti_varianta, test_otazka_id, test_otazka_spravne)
Tridni_kniha	(tridni_kniha_id, rozvrh_id, tridni_kniha_popis, tridni_kniha_datum, tridni_kniha_datum_vyplneni, zamestnanec_id, rok_id)
Vypisy_znamek	(vypis_id, trida_id, zak_id, vypis_datum_od, vypis_datum_do, vypis_datum)

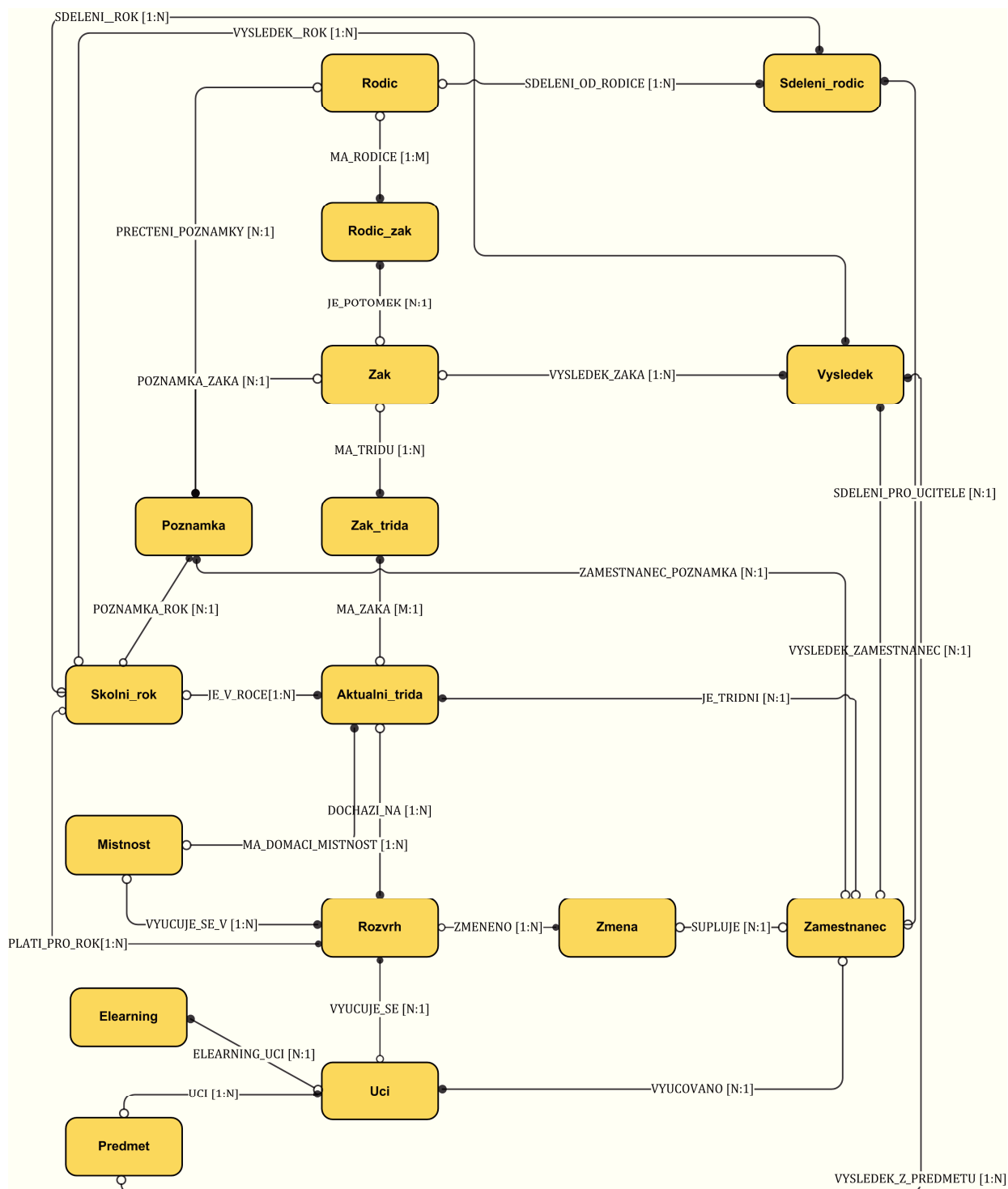
3.2 Popis vztahů:

MA_RODICE	(Rodice, Rodice_zak)	1:N
JE_POTOMEK	(Rodice_zak, Zak)	N:1
MA_TRIDU	(Zak, Zak_trida)	1:N
MA_ZAKA	(Zak_trida, Aktualni_trida)	M:1
DOCHAZI_NA	(Aktualni_trida, Rozvrh)	1:N
VYUCUJE_SE	(Rozvrh, Uci)	N:1
VYUCOVANO	(Uci, Zamestnanec)	N:1
VYUCUJE_SE_V	(Rozvrh, Mistnost)	1:N
MA_DOMACI_MISTNOST	(Trida, Mistnost)	1:N
JE_TRIDNI	(Aktualni_trida, Zamestnanec)	N:1
ZMENENO	(Rozvrh, Zmena)	1:N
SUPLUJE	(Zamestnanec, Zmena)	N:1
JE_V_ROCE	(Skolni_rok, Aktualni_trida)	1:N
UCI	(Uci, Predmet)	N:1
PLATI_PRO_ROK	(Skolni_rok, Rozvrh)	1:N
VYSLEDEK_ZAKA	(Zak, Vysledek)	1:N
VYSLEDEK_Z_PREDMETU	(Predmet, Vysledek)	1:N
VYSLEDEK_ZAMESTNANEC	(Vysledek, Zamestnanec)	N:1
VYSLEDEK_ROK	(Vysledek, Skolni_rok)	N:1
POZNAMKA_ZAKA	(Poznamka, Zak)	N:1
PRECTENI_POZNAMKY	(Poznamka, Rodice)	N:1
ZAMESTNANEC_POZNAMKA	(Zamestnanec, Poznamka)	1:N
POZNAMKA_ROK	(Skolni_rok, Poznamka)	1:N
SDELENI_OD_RODICE	(Rodice, Sdeleni_rodice)	1:N
SDELENI_ROK	(Skolnic_rok, Sdeleni_rodice)	1:N
SDELENI_PRO_UCITELE	(Sdeleni_rodice, Zamestnanec)	N:1
ELEARNING_UCI	(Elearning, Uci)	N:1

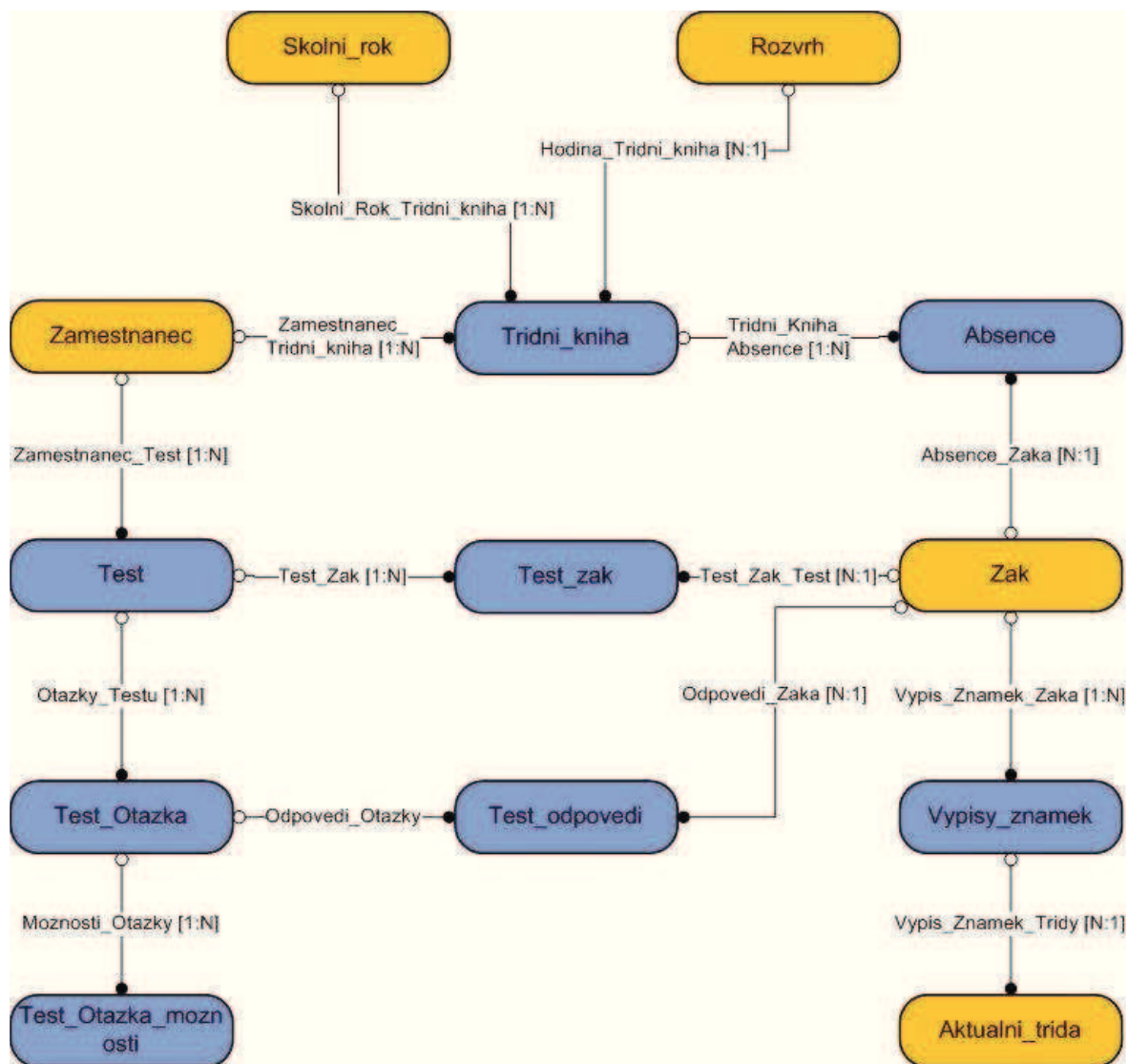
Nové vztahy:

HODINA_TRIDNI_KNIHA	(Tridni_kniha, Rozvrh)	N:1
SKOLNI_ROK_TRIDNI_KNIHA	(Skolni_rok, Tridni_kniha)	1:N
ZAMESTNANEC_TRIDNI_KNIHA	(Zamestnanec, Tridni_kniha)	1:N
TRIDNI_KNIHA_ABSENCE	(Tridni_kniha, Absence)	1:N
ZAMESTNANEC_TEST	(Zamestnanec, Test)	1:N
OTAZKY_TESTU	(Test, Test_otazka)	1:N
MOZNOSTI_OTAZKY	(Test_otazka, Test_otazka_moz)	1:N
TEST_ZAK	(Test, Test_zak)	1:N
TEST_ZAK_TEST	(Test_zak, Zak)	N:1
ODPOVEDI_ZAKA	(Test_odpovedi, Zak)	N:1
VYPIS_ZNAMEK_ZAKA	(Zak, Vypisy_znamek)	1:N
VYPIS_ZNAMEK_TRIDY	(Vypisy_znamek, Akt_trida)	N:1
ABSENCE_ZAKA	(Absence, Zak)	N:1

3.3 ERD



Obrázek č. 2: ER diagram pro databázi z bakalářské práce



Obrázek č. 3: ER diagram pro nově implementované databázové tabulky

3.4 Datový slovník

[primární klíč, cizí klíč (FK)]

V této části je uveden pouze datový slovník pro nově vytvořené databázové tabulky. Kompletní datový slovník je uveden v příloze D.

Absence					
Název	Datový typ (velikost)	Klíč	Null	Index	IO, popis
absence_id	Integer	A	N	A	Auto increment
zak_id	Integer	N	N	A	FK – Zak
tridni_kniha_id	Integer	N	N	A	FK – Tridni kniha
absence_popis	Text	N	N	N	

Test					
Název	Datový typ (velikost)	Klíč	Null	Index	IO, popis
test_id	Integer	A	N	A	Auto increment
test_nazev	Varchar(100)	N	N	N	
test_popis	Text	N	A	N	
zamestnanec_id	Integer	N	N	A	FK – Zamestnanec
test_vytvoril_zamestn_id	Integer	N	N	A	
test_od	Datetime	N	N	N	
test_do	Datetime	N	N	N	
test_zak_id	Integer	N	A	A	FK – Test_zak

Test_odpovedi					
Název	Datový typ (velikost)	Klíč	Null	Index	IO, popis
test_odpovedi_id	Integer	A	N	A	Auto increment
otazka_id	Integer	N	N	A	FK – Test_otazka
zak_id	Integer	N	N	A	FK - Zak
test_odpoved	Text	N	N	N	

Test_otazka					
Název	Datový typ (velikost)	Klíč	Null	Index	IO, popis
test_otazka_id	Integer	A	N	A	Auto increment
test_otazka_text	Text	N	N	N	
test_otazka_poznamka	Text	N	A	N	
test_id	Integer	N	N	A	FK – Test

Test_otazka_moznosti					
Název	Datový typ (velikost)	Klíč	Null	Index	IO, popis
test_otazka_moznosti_id	Integer	A	N	A	Auto increment
test_otazka_m. varianta	Varchar(45)	N	N	A	
test_otazka_id	Integer	N	N	N	FK – Test_otazka
test_otazka_spravne	Varchar(100)	N	A	N	

Test_zak					
Název	Datový typ (velikost)	Klíč	Null	Index	IO, popis
test_zak_id	Integer	A	N	A	Auto increment
zak_id	Integer	N	N	A	FK – Zak
test_id	Integer	N	N	A	FK – Test

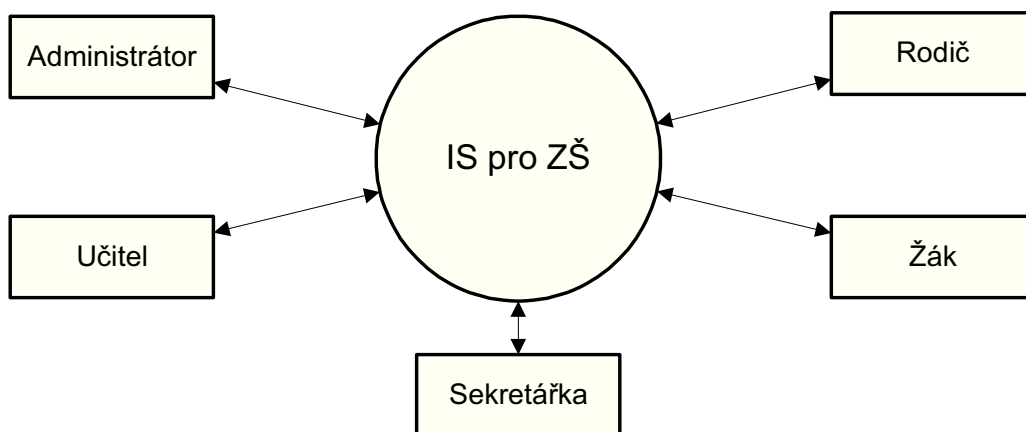
Tridni_kniha					
Název	Datový typ (velikost)	Klíč	Null	Index	IO, popis
tridni_kniha_id	Integer	A	N	A	Auto increment
rozvrh_id	Integer	N	N	A	FK – Rozvrh
tridni_kniha_popis	Text	N	A	N	
tridni_kniha_datum	Date	N	A	A	
tridni_kniha_dat. vytvoreni	Date	N	A	A	
zamestnanec_id	Integer	N	N	A	FK – Zamestnanec
rok_id	Integer	N	N	A	FK – Skolni_rok

Vypisy_znamek					
Název	Datový typ (velikost)	Klíč	Null	Index	IO, popis
vypis_id	Integer	A	N	A	Auto increment
trida_id	Integer	N	A	A	FK – Aktualni_trida
zak_id	Integer	N	A	A	FK - Zak
vypis_datum_od	Date	N	N	N	
vypis_datum_do	Date	N	N	N	
vypis_datum	Date	N	N	N	

4 Funkční analýza

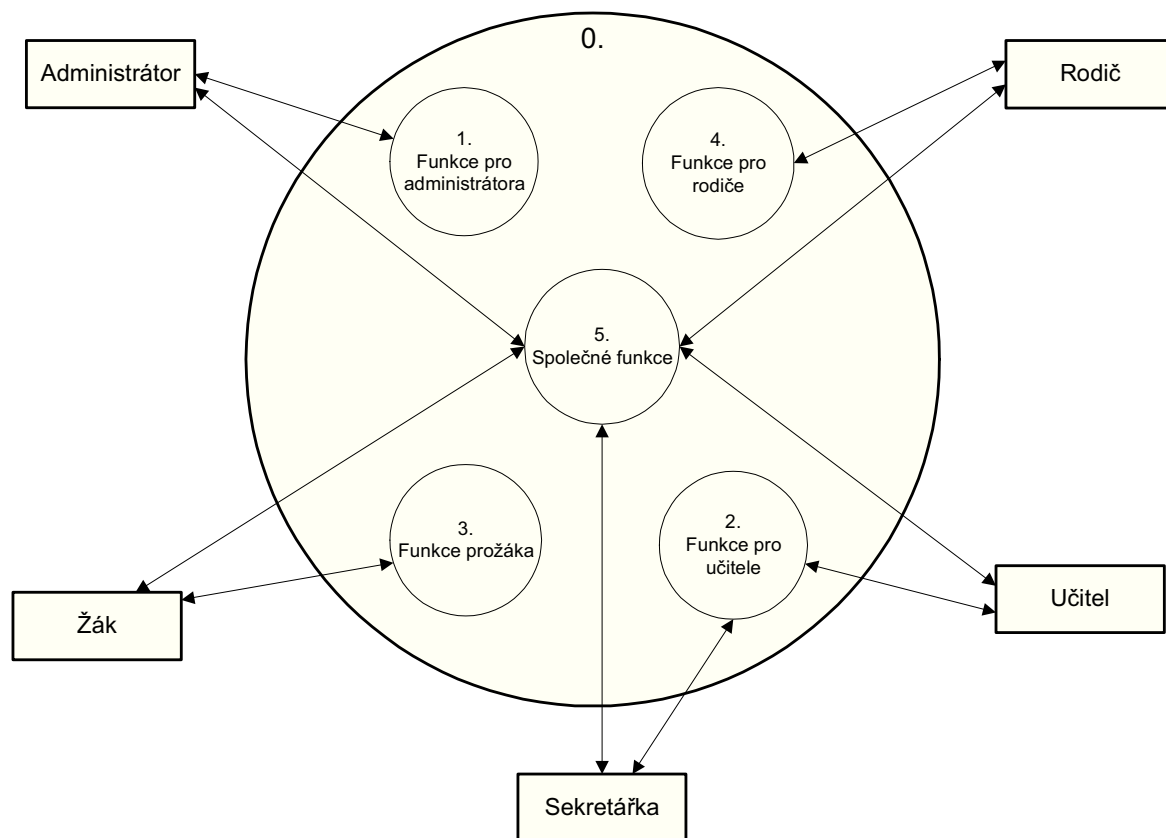
Vzhledem k tomu, že většina nově implementovaných funkcí se týká učitele, je i následující analýza zaměřena na tyto funkce. V následující části bude uveden DFD 0. úrovně, DFD 1. úrovně pro funkce učitele a dále budou rozepsány jednotlivé nové funkce týkající se převážně učitele. Vzhledem k charakteru implementovaných funkcí mají ostatní role uživatelů přístup jen k výpisům těchto funkcí. Zde se převážně jedná o výpisy například absence testových otázek. U těchto funkcí se není třeba zabývat dopodrobna jejich analýzou.

4.1 Kontextový diagram



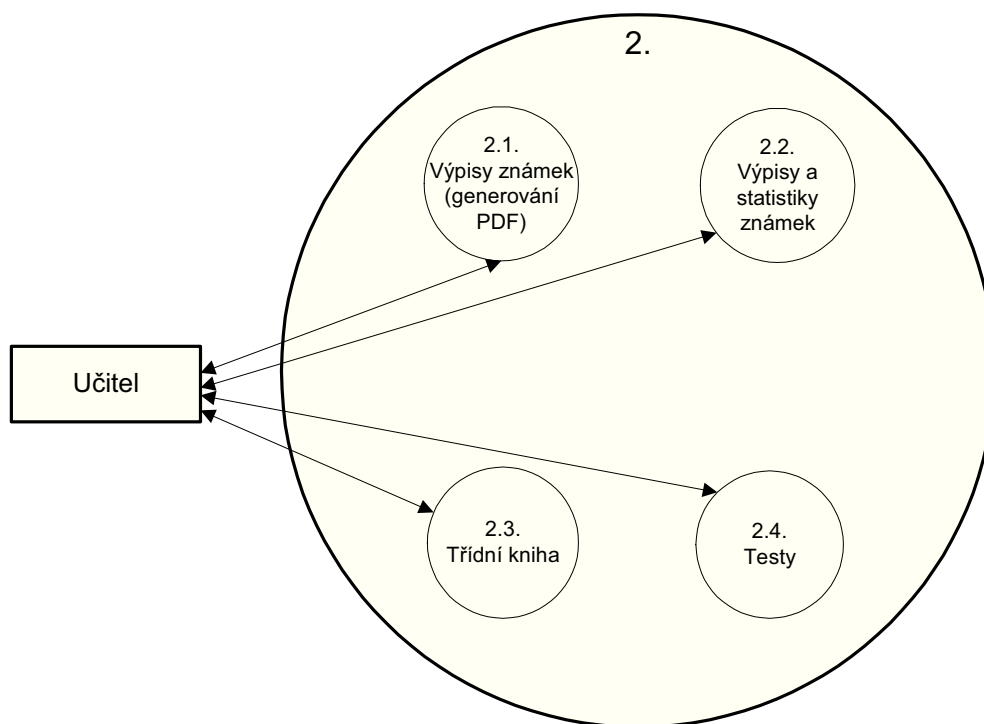
Obrázek č. 4: Kontextový diagram

4.2 DFD 0. úrovně



Obrázek č. 5: DFD 0. Úrovně

4.3 DFD 1. úrovně - Funkce pro učitele

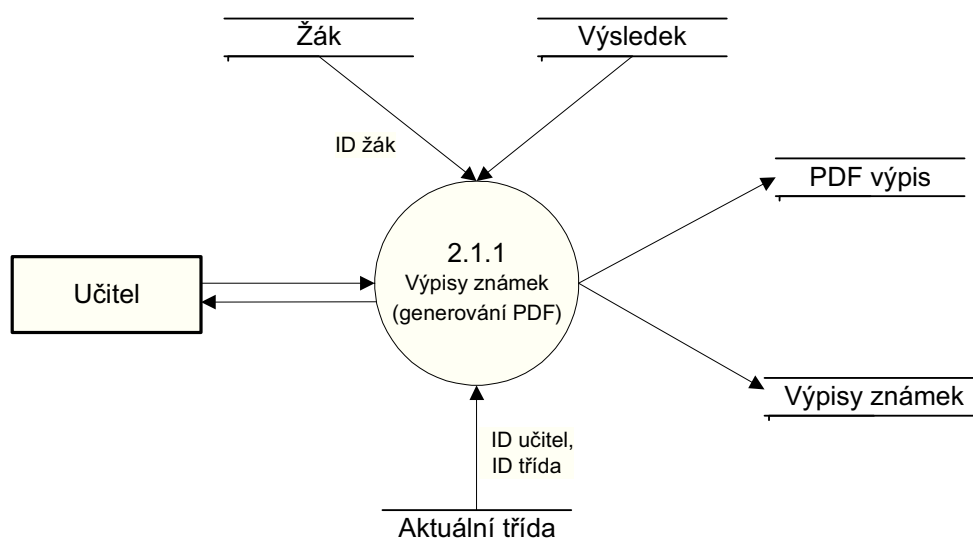


Obrázek č. 6: DFD 1. Funkce pro učitele

4.4 Minispecifikace

V následujících kapitolách budou uvedeny minispecifikace hlavních funkcí IS. Funkce, které výrazně zasahují nebo mění již implementované a používané části IS, jsou blíže popsány v kapitole Popis implementace, kde jsou zejména popsány změny již hotových částí IS a začlenění nových funkcí do stávajícího prostředí

4.4.1 Vytvoření výpisu známek

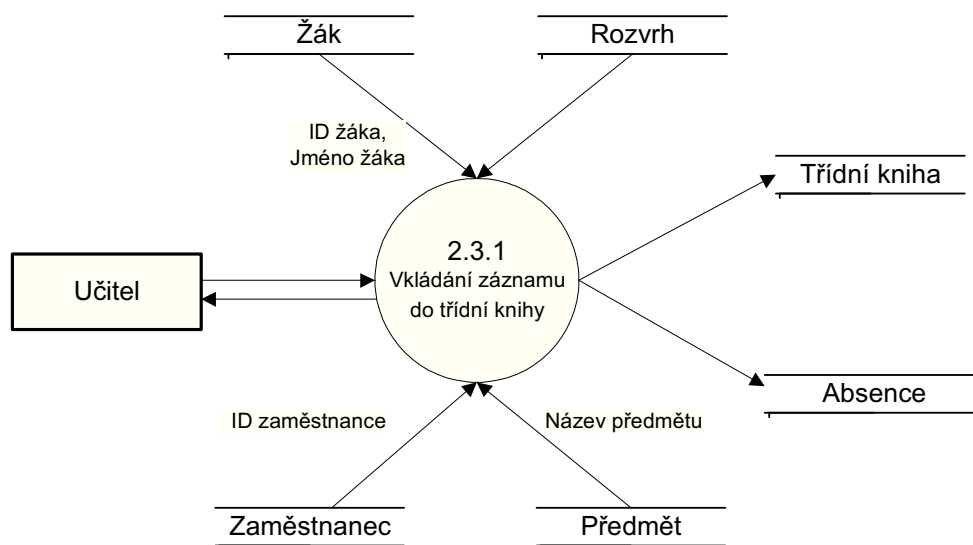


Obrázek č. 7: DFD pro vytvoření výpisu známek

1. Zobraz formulář pro výběr mezi výpisem známek pro žáka a výpisem známek pro celou třídu,
2. učitel vybere, jestli chce generovat výpisy pro jednotlivého žáka nebo pro celou třídu,
3. zobraz seznam vygenerovaných výpisů známek podle tabulky *vypisy_znamek*,
4. zobraz formulář pro výběr počátečního a koncového data výpisu známek,
5. uživatel zvolí požadovaný rozsah data,
6. v případě volby výpisu pro celou třídu proved' následující kroky:
 - a. načti do formuláře seznam tříd,
 - b. učitel vybere požadovanou třídu,
 - c. z tabulky *vysledek* načti výsledky pro žáky z vybrané třídy pro daný časový rozsah,
7. v případě volby výpisu pro jednotlivé žáky proved' následující kroky:
 - a. načti do formuláře seznam tříd,
 - b. uživatel vybere požadovanou třídu,
 - c. načti do formuláře seznam žáků vybrané třídy,
 - d. učitel vybere žáka, pro kterého chce generovat výpis známek,
 - e. načti z tabulky *vysledek* výsledky pro zvoleného žáka pro daný časový rozsah,

8. z tabulky *zak* načti jméno a příjmení žáka,
9. ze jména žáka, názvu zvolené třídy a školního roku podle tabulky *skolni_rok* vytvoř hlavičku výpisu známek,
10. z hlavičky dokumentu získané v kroku 6. a z načtených výsledků pro žáka nebo více žáků dané třídy vygeneruj pdf soubor s celkovým výpisem známek,
11. vlož do tabulky *vypis_znamek* záznam o vygenerovaném výpisu,
12. zobraz pdf soubor s výpisem učiteli.

4.4.2 Vložení záznamu do třídní knihy

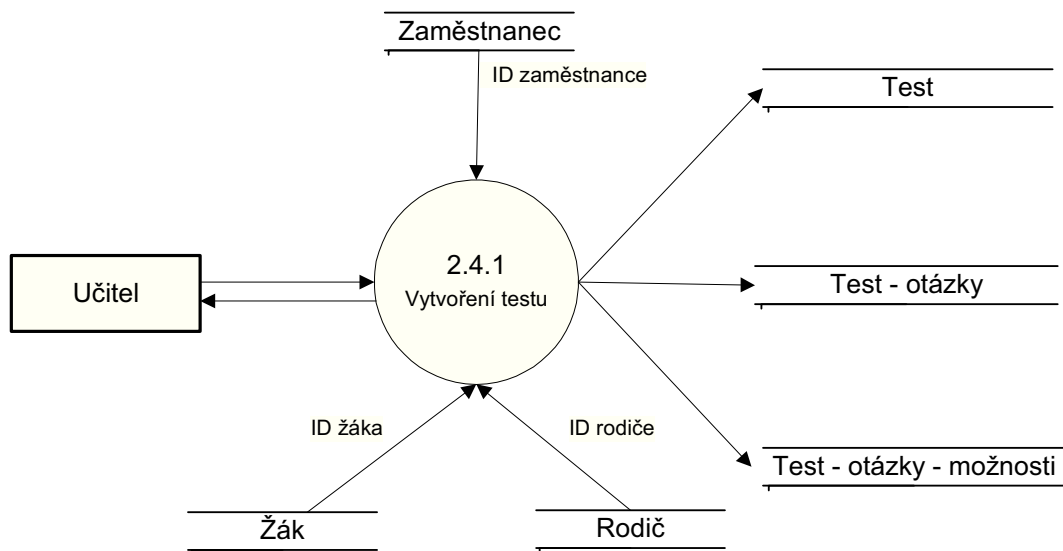


Obrázek č. 8: DFD pro vložení záznamu do třídní knihy

1. Zobraz formulář pro vkládání záznamů do třídní knihy a do sloupce předmět načti předměty podle tabulky *rozvrh* a jejich názvy z tabulky *predmet*
2. učitel vyplní probírané učivo do příslušného sloupce,
3. po potvrzení učitelem vlož záznam do tabulky *tridni_kniha*, jméno vyučujícího je vloženo automaticky podle přihlášeného učitele z důvodu bezpečnosti, učitel však může zapisovat i údaje o učivu pro předměty, které nevyučuje,
4. v případě, že někteří žáci chybí ve vyučování, učitel vybere položku absence,
5. zobraz nový formulář a načti do něj seznam žáků příslušné třídy podle tabulek *aktualni_trida* a *zak*,
6. učitel vybere chybějící žáky a potvrdí zadané údaje,
7. vlož hodnoty vyplněné učitelem do tabulky *absence*,
8. zobraz zpět formulář pro zadávání údajů do třídní knihy z kroku 1. a do sloupce absence načti nově vyplněné hodnoty podle tabulky *absence*,
9. učitel může dále pokračovat s prací s formulářem od kroku 1.

Datum	Předmět	Učivo	Vyučující	Absence
12.1.2011 1. hodina	M	Zlomky – opakování	Jan Novák	Čech, Novák
12.1. 2011 2. hodina	ČJ	Gramatika – test	Karel Petřík	Čech
12.1. 2011 3. hodina	TV	Fotbal	Petr Nový	Čech

4.4.3 Vytvoření testu



Obrázek č. 9: DFD pro vložení záznamu do třídní knihy

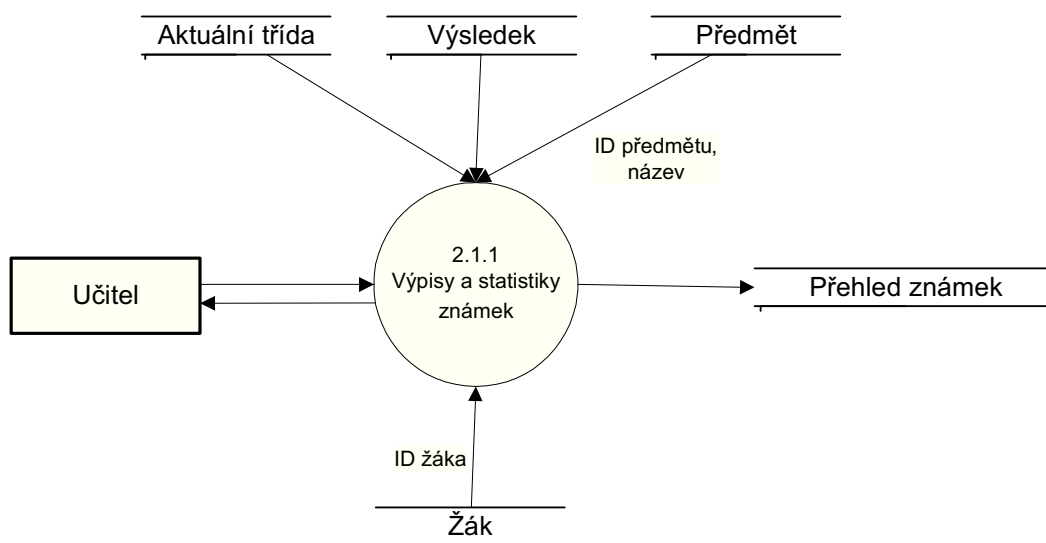
1. Zobraz formulář s výpisem vytvořených testů pro přihlášeného učitele podle tabulky *test*, a volbu pro vytvoření nového testu,

Datum vytvoření	Dostupný od	Dostupný do	Název	Výsledky	Editace
12. 2. 2011	14. 2. 2011	15. 2. 2011	Čtvrtletní test z matematiky	Zobrazit	Editovat
13.2.2011	15.3. 2011	16.3.2011	Opakování gramatiky	Zobrazit	Editovat

2. učitel zvolí vytvoření nového testu,
3. zobraz formulář pro výběr uživatelů, pro které bude test určen s možnostmi žák, třída, rodič,
4. učitel vybere skupinu uživatelů, pro které chce zpřístupnit vytvořený test,
5. podle zvolené skupiny uživatelů načti seznam uživatelů z tabulky *zak*, *aktualni_trida* nebo *rodic*,
6. zobraz formulář pro vytvoření testu se vstupními poli pro datum dostupnosti testu a název testu,
7. učitel vyplní formulář,
8. zapiš údaje z formuláře do tabulky *test*, a do tabulky *test_zak* zapiš id uživatelů, pro které má být test dostupný,

9. zobraz formulář pro vytvoření testové otázky, s volbou počtu a typu odpovědí,
10. učitel postupně vytváří jednotlivé otázky testu,
11. vlož vyplněné údaje do tabulek *test_otazka* a *test_moznosti*,
12. pokračuj na úvodní seznam testů z kroku 1.

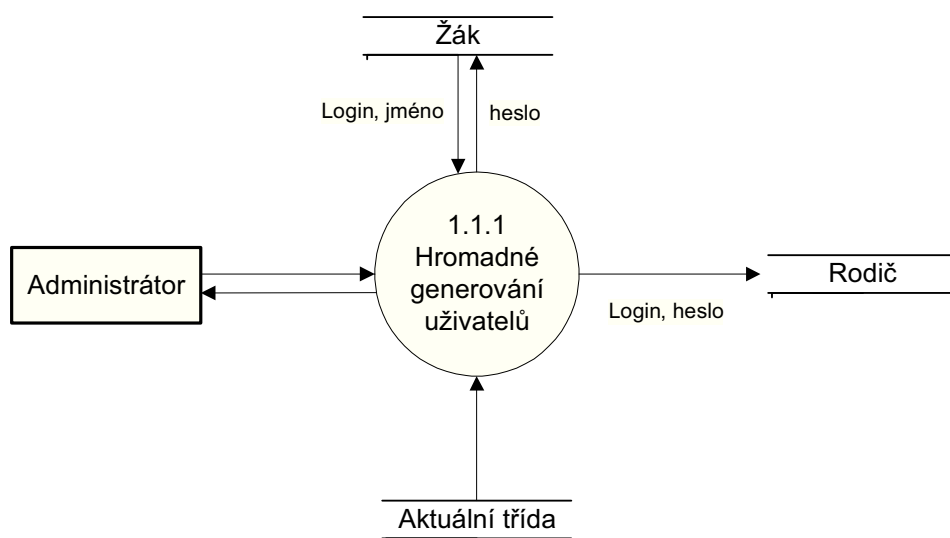
4.4.4 Zobrazení výpisu známek



Obrázek č. 10: DFD pro zobrazení výpisu známek

1. Zobraz formulář pro výběr třídy, do formuláře načti třídy, ve kterých učitel vyučuje alespoň jeden předmět podle tabulek *zamestnanec*, *uci* a *aktuální_trida*,
2. učitel vybere třídu, pro kterou chce zobrazit výpis,
3. zobraz formulář a načti do něj seznam předmětů, které přihlášený učitel vyučuje ve výše vybrané třídě podle tabulek *uci*, *predmet* a *aktuální_trida*,
4. zobraz formulář pro výběr data výpisu,
5. učitel vybere předmět, pro který chce zobrazit výpis známek a rozsah data pro výpis,
6. z tabulky *zak* vypiš seznam žáků zvolené třídy a z tabulky *vysledek* pro každého žáka vypiš jeho známky z výše vybraného předmětu, jednotlivé váhy známek vypiš odlišným stylem,
7. pro vypsání známek vypočítej hodnoty jejich váženého průměru pro jednotlivé žáky i celou třídu a dále pak počty známek pro žáky a medián.

4.4.5 Hromadné generování uživatelů



Obrázek č. 11: DFD pro hromadné generování uživatelů

1. Zobraz formulář pro výběr třídy,
2. načti do formuláře názvy všech tříd pro aktuální období z tabulky *aktualni_trida*,
3. administrátor vybere třídu, pro kterou chce generovat rodiče a žáky,
4. vypiš seznam žáků ze zvolené třídy podle tabulek *aktualni_trida* a *zak*,
5. administrátor potvrdí seznam žáků stisknutím tlačítka generovat,
6. podle tabulek *aktualni_trida* a *zak*, postupně pro každého žáka zvolené třídy proved' následující operace,
7. pro žáka vygeneruj heslo ve formě náhodného pěticiferného čísla a ulož jej do pole *zak_heslo* v tabulce *zak*,
8. v tabulce *rodic* přidej pro žáka dva záznamy, kde *rodic_login* bude ve tvaru *login_zaka+r1* a *login_zaka+r2* a v polích *rodic_heslo* bude opět náhodné pěticiferné číslo,
9. do vazební tabulky *rodic_zak* přidej příslušné vazební údaje,
10. po vygenerování uvedených údajů pro všechny žáky zvolené třídy vypiš údaje ve formě:

Jméno žáka:	Martin Novák
Login žáka:	novama
Heslo žáka:	12345
Login rodiče 1	novamar1
Heslo rodiče 1	12345
Login rodiče 2	novamar2
Heslo rodiče 2	12345

5 Návrh implementace

První část IS, která byla implementována v rámci dvou výše uvedených bakalářských prací, používala kombinaci PHP a MySQL. S tímto řešením nebyly během implementace ani provozu zaznamenány žádné závažnější problémy. Ukázalo se však, že implementace některých funkcí (ověřování uživatelských vstupů, tvorba formulářů) je zbytečně pracná a některé rozsáhlé části IS začínají být nepřehledné. Z důvodu předcházení těmto komplikacím byl vybrán pro implementaci dalších částí IS framework NETTE, který bude dále detailněji popsán.

5.1 Upřesnění požadovaných SW a HW technologií

Jak bylo uvedeno výše, vývoj nových částí IS bude probíhat v technologiích PHP a MySQL. Během doby implementace a používání stávajícího IS se obě technologie postupně vyvíjely, proto je pro další vývoj třeba použít nejnovější verze uvedených technologií. Jako framework pro PHP bude sloužit framework NETTE v aktuální verzi.

IS bude provozován na externím webhostingu na operačním systému Linux a samotný vývoj bude probíhat v prostředí MS Windows.

5.2 Popis jednotného konfiguračního souboru

Při implementaci IS v rámci bakalářské práce bylo cílem vytvořit obecný IS, který by byl jednoduše použitelný na více školách. Toto se dle mého názoru podařilo. Při spouštění IS na jiných školách se však ukázalo, že bylo poměrně obtížné zadat všechna počáteční nastavení pro spuštění jako je doména školy, přístupové údaje k databázi, název školy, váhy známek atd. Proto bylo rozhodnuto o vytvoření jednoho souboru, který bude obsahovat všechna nastavení IS. Díky tomu bude stačit při prvotní instalaci IS nastavit vše přehledně v jednom konfiguračním souboru.

5.3 Výkon

Dle statistik návštěvnosti stránek měly www stránky a IS školy Karla Dvořáčka denní návštěvnost cca 200 návštěvníků. Zde je zřejmé, že i přesto, že celkový počet uživatelů se blíží dvěma tisícům, tak je používání IS rozloženo v průběhu celého dne (obvykle dopoledne zadávání známek učiteli, odpoledne prohlížení známek rodiči). Vzhledem k uvedenému rozložení používání IS a k dostatečnému výkonu použitých webhostingových služeb nebyly zaznamenány žádné problémy s výkonem. Výkon IS byl sledován jednak vlastním testováním při práci s administračním rozhraním, tak prostřednictvím zkušeností uživatelů.

Dále byla sledována výkonnost použité databáze. Zde opět nebyly žádné problémy. Roční objem dat se pohybuje kolem cca 15 MB, což není žádný problém vzhledem ke kapacitě databáze. Jediný problém může po více letech představovat tabulka výsledek, kde se po dvou letech provozu nachází cca 30 000 záznamů. Zde by patrně bylo vhodné po určité době odstraňovat starší záznamy.

Pro optimalizaci rychlosti databáze byly podrobně navrženy jednotlivé indexy, které jsou uvedené v datovém slovníku v kapitole Datová analýza.

Během více než dvou let provozu nebyly zaznamenány žádné závažnější technické problémy jak s provozem webhostingu, tak s provozem domény. Jediný problém, se vyskytl během upgradu webhostingových serverů firmy Forpsi na clusterové řešení, po kterém přestaly fungovat na ZŠ Karla Dvořáčka některé části IS, které využívaly soubory (například css styly), na které bylo odkazováno v aplikaci bez „www“ v adrese souboru. Tento problém byl však způsoben nestandardní konfigurací domény, kde je adresa www.zsdvoracka.cz směřována na servery firmy Forpsi.cz a adresa zsdvoracka.cz je směřována na externí poštovní server. Po upgradu serverů se však změnilo chování některých odkazů, což zapříčinilo nefunkčnost souborů odkazovaných prostřednictvím adresy bez www. Tento problém byl však po konzultaci s technickou podporou rychle odstraněn.

Některé další změny v konfiguraci, které byly řešeny s technickou podporou, byly vždy bezproblémově vyřešeny. Veškeré problémy se týkaly jen prvního spouštění a převodu domény tak, jak bylo popsáno v [1].

5.4 Zálohování

Během zhruba dvouletého provozu se neobjevil žádný problém s technickou stránkou webhostingu, což svědčí o vysoké úrovni poskytovatele webhostingových služeb firmě Internet CZ, která provozuje webhosting Forpsi. I přesto se však doporučuje v pravidelných týdenních intervalech provádět zálohu databáze. Všechny údaje v databázi se sice uchovávají i po vymazání uživatelem, ale pro případ nějaké nepředvídatelné situace je vždy vhodné mít dostupnou aktuální zálohu databáze.

S tímto postupem při zálohování jsou seznámeni příslušní ICT metodici z uvedených škol.

Z důvodu bezpečnosti a možnosti uchovávat starší výsledky byla provedena záloha na konci každého pololetí.

5.5 Uživatelské rozhraní

Při počátečním návrhu uživatelského rozhraní byl kladen důraz na co největší jednoduchost použití. Tento přístup se velice osvědčil. Po grafické stránce se jedná o jednoduché rozhraní bez zbytečných prvků. O to větší důraz byl kladen na samotnou funkčnost. Průběžně byly analyzovány reakce uživatelů a rozhraní bylo upravováno s ohledem na minimalizaci doby pro přístup k požadovaným položkám. Zvláště důležitý byl návrh administračního rozhraní, kde například při importech dat žáků každé kliknutí navíc značně zpomaluje práci. Z tohoto důvodu byly také implementovány některé administrativní úkoly, jako třeba importy žáků a generování jejich hesel ve formě hromadných operací, které zbytečně administrátory nezatěžují rutinní prací.

Osvědčila se také nápověda, která byla vkládána ke složitějším funkcím a poskytovala tak rychle dostupný popis dané funkce a jejího ovládání.

Uživatelské rozhraní bylo také přizpůsobeno on-line provozu. V praxi to znamená, že učitelé budou mít dostupný vlastní počítač a výsledky žáků a údaje do třídní knihy budou moci zadávat

přímo během vyučování. V tomto ohledu byly analyzovány požadavky uživatelů a ukázalo se, že současný způsob zadávání známek jim i nadále vyhovuje a jediným požadavkem je přizpůsobení třídní knihy, která se bude vypisovat podle aktuálního kalendářního dne.

6 Implementace

6.1 HW/SW

Jak již bylo zmíněno v kapitole Výkon, použité technické řešení představované webhostingem DYNAMIChosting na OS Linux se osvědčilo a nebylo třeba nic měnit. Stejná konfigurace od stejné společnosti byla použita i pro ZŠ Karla Dvořáčka.

Použitý software

Operační systém	MS Windows 7
WWW server	Apache 2.2.17
Verze PHP	PHP 5.3.3
SŘBD	MySQL 5.5.8
Administrace DB	MySQL Administrator 1.2.12 MySQL Query Browser 1.2.12 phpMyAdmin 3.2.0.1
Verze NETTE	0.9.7
Textový editor	PSPad 4.5.4 MS Word 2007
Kreslení schémat	MS Visio 2007
Dokumentace	phpDocumentor v1.4.3

6.2 Popis frameworku NETTE

Framework NETTE je dílem českého autora Davida Grudla, který jej vytvořil zpočátku pro své osobní použití. Brzy se však framework začal volně šířit a rychle si získal velkou oblibu u mnoha vývojářů. Toto jistě svědčí o jeho vysoké kvalitě. Během pár let se z nové technologie určené jen pro osobní použití stal plnohodnotný framework, na kterém dnes běží například [www stránky serveru Root¹](#), stránky nové scény národního divadla², prezidenta České republiky³, mladé fronty⁴, zpravodajského serveru E15⁵ a mnoho dalších.

Protože se jedná o relativně novou technologii a navíc od českého autora, bude zde popsána struktura tohoto frameworku a dále budou blíže popsány některé jeho vlastnosti.

¹ www.root.cz

² www.novascena.cz/cs

³ www.klaus.cz

⁴ www.mfplus.cz

⁵ www.e15.cz

- **MVC – Model – View – Controller**
- **Licence**
- **Zabezpečení**
- **XSS - Cross-site scripting**
- **CSRF - Cross-site request forgery**
- **Zneužití sessions**
- **Ladící nástroje**
- **Výkon**
- **Komunita**
- **Doporučené postupy**
- **AJAX**
- **DRY**
- **KISS**

- **MVC – Model – View – Controller**

Jedná se o softwarovou architekturu, kde je aplikace rozdělena na datový model - model, řídicí logiku – controller a uživatelské rozhraní – view. Ve frameworku NETTE se místo části view používá označení presenter. Detailní popis MVC je nad rámec této práce. Blíže je popsán například v [12]. MVP používaný ve frameworku NETTE bude popsán v samostatné kapitole.

- **Licence**

NETTE je svobodný software. To znamená, že uživatel má přístup ke zdrojovým souborům frameworku a může s nimi libovolně zacházet. Může tedy mimo standardního používání jakékoli části libovolně distribuovat a měnit. Tato licence jednak umožňuje zkušenějším vývojářům v případě potřeby měnit jednotlivé součásti dle potřeby a dále řeší situaci, kdy by původní autor přestal framework dále vyvíjet. V tomto případě se může dalšího vývoje ujmout kdokoli jiný, případně se o něj může starat celá uživatelská komunita. Celá licence je uvedena v [13].

- **Zabezpečení**

NETTE klade velký důraz na bezpečnost aplikací. Mezi hlavní způsoby, které poskytuje, patří následující:

- **XSS - Cross-site scripting**

Jedná se o metodu, kdy útočník podstrčí vlastní kód prostřednictvím neošetřených vstupů webové aplikace, což mu umožní získat přístup k citlivým informacím nebo vyvolat neočekávané chování aplikace. NETTE toto řeší technologií Context-aware escaping, která vylepšuje automatickou kontrolu vstupních dat běžnou u jiných frameworků o zohlednění původu dat. Jiný postup se tedy používá například pro javascript a jiný pro html kód. Toto rozpoznávání je řešeno automaticky.

- **CSRF - Cross-site request forgery**

Tento útok (také známý pod pojmem one-click attack) spočívá v tom, že útočník donutí uživatele navštívit stránku, která poté zneužije toho, že je uživatel v nějaké aplikaci přihlášen a může zneužít jeho identitu pro provedení škodlivé akce.

Ochrana proti tomuto typu útoku spočívá ve vygenerování náhodného tokenu, jehož hodnotu útočník nezná a nemůže jej tak podstrčit. Takto lze jednoduše ověřit, zda se jedná o skutečný požadavek aplikace.

- **Zneužití sessions**

U tohoto druhu útoku útočník buď podstrčí, nebo odcizí session ID a tím získá přístup k aplikaci. NETTE tomu zabráňuje tím, že automaticky konfiguruje PHP, čímž zajistí zabezpečení před těmito útoky, protože tento typ útoku je způsobován chybnou konfigurací PHP.

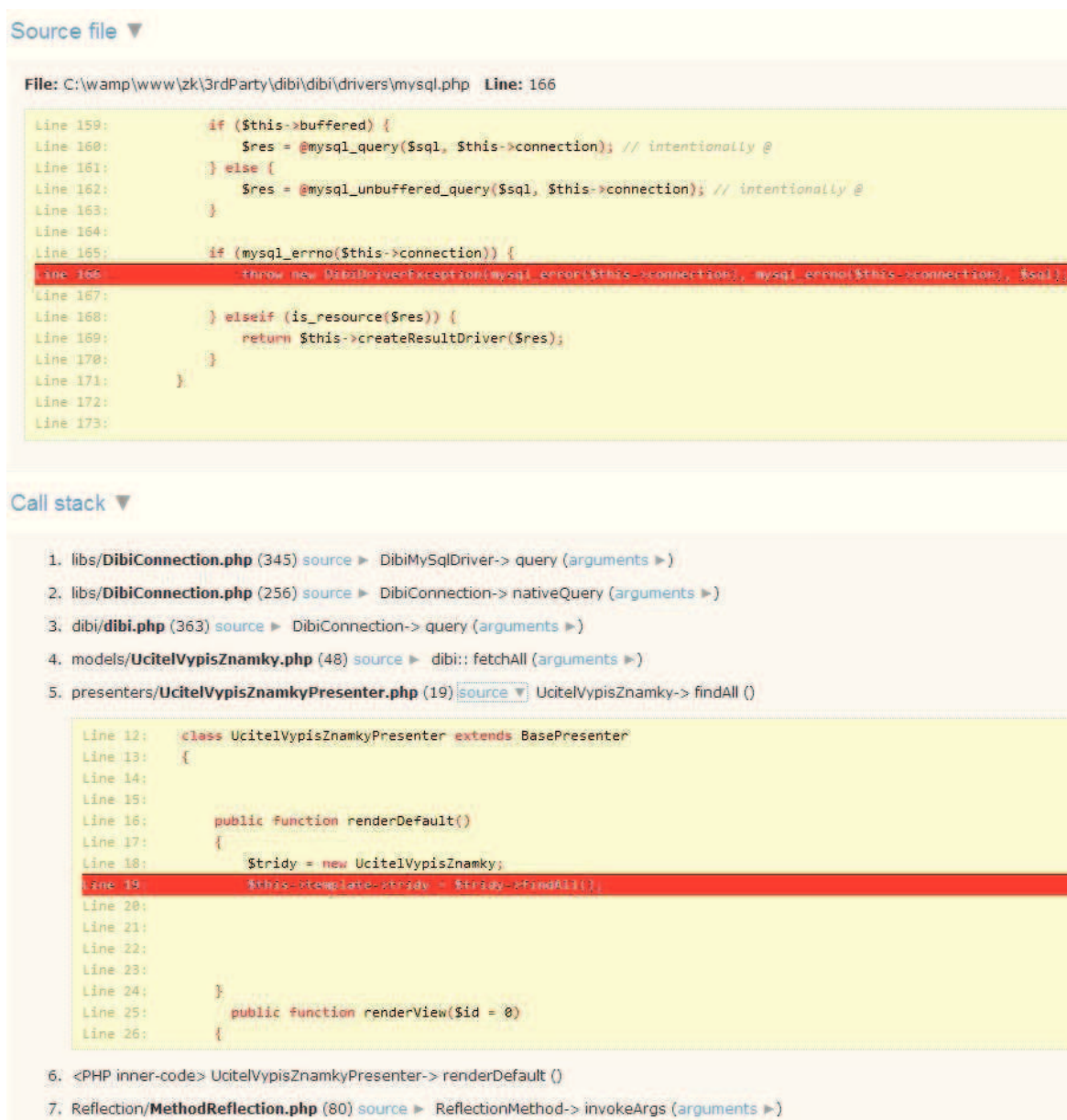
- **Ladící nástroje**

Vzhledem k tomu, že základní verze PHP nepodporuje žádné rozšíření pro ladění aplikace, je toto řešeno frameworkem Nette. Ten poskytuje přehledné informace o chybě a tím výrazně usnadňuje hledání chyb.

V kódu aplikace zapneme ladění aplikace příkazem:

```
Debug::enable();
```

Ladící režim také rozlišuje mezi vývojovým a produkčním prostředím pro vývoj aplikací. Tomu jsou přizpůsobeny výpisy chyb, které se ve vývojovém prostředí vypisují přímo v internetovém prohlížeči, zatímco v produkčním prostředí se nezobrazují uživatelům, ale zapisují se do souboru. Ve výchozím nastavení se jedná o soubor `php_error.log`. Je také možné nastavit upozorňování o chybách prostřednictvím emailových zpráv.



Obrázek č. 12: Ukázka ladícího výpisu NETTE frameworku

Na níže uvedeném obrázku č. 12 vidíme ukázku ladícího výpisu poskytovaného frameworkem Nette. Ladící režim také spolupracuje s doplňkem Firebug pro internetový prohlížeč Firefox. Toto je však již nad rámec této práce. Podrobnější informace lze nalézt v [14].

- **Výkon**

Podle testu uvedeného na serveru Root.cz [15] je NETTE jedním z nejrychlejších frameworků pro PHP.

- **Komunita**

Velkou výhodou NETTE je početná komunita vývojářů. Mnoho z nich je navíc z České republiky, což zajistí příjemnou vzájemnou komunikaci. V rámci komunity jsou vytvářeny různé doplňky, pluginy a další rozšíření. Součástí komunitních stránek je také nezbytné diskusní fórum. Zde je výhodou, že jedním z hlavních autorů na fóru je samotný autor NETTE David Grudl, který tak na základě reakcí a požadavků uživatelů průběžně framework aktualizuje.

- **Doporučené postupy**

NETTE framework vychází z ověřených postupů pro návrh aplikací. Programátora tímto vede k používání objektového programování a používání technologií AJAX, SEO, DRY, KISS, MVC, které NETTE podporuje a které budou dále detailněji popsány.

- **AJAX**

V roce 2005 byl poprvé zaveden termín AJAX, který označuje technologie a postupy při tvorbě interaktivních webových aplikací.

Jedná se o technologie (X)HTML a CSS pro prezentaci informací, JavaScript pro dynamické chování aplikací a (XML)HttpRequest pro výměnu informací mezi klientem a serverem. Na rozdíl od standardních webových prezentací je tímto umožněno interaktivní chování www stránek, kdy není potřeba znovu načítat ze serveru celou stránku při každé změně. Typicky je pak touto technologií řešeno například stránkování výpisů položek, hlasování v anketách, vyhledávání a podobně.

- **DRY**

Tento princip je obdobou modelem řízené architektury (MDA). U MDA se jedná o oddělení business a aplikační logiky od technologické platformy. Princip DRY spočívá ve snaze zamezit opakování informací na více místech v aplikaci. Vývojář tak není nucen pracovat metodou „copy - paste“ a každá informace, nastavení a podobně jsou vždy jen na jednom místě ve zdrojovém kódu. Díky tomu je aplikace přehlednější a méně náročná na údržbu. V praxi se tento postup projevuje tak, že pokud se změní jednotlivá část aplikace (systému), nejsou ovlivněny zbývající části a pokud jsou prvky logicky strukturovány, má jedna změna vliv na celou příslušnou skupinu.

- **KISS**

Tento přístup označuje snahu o co nejjednodušší a nejpréhlednější kód aplikace. Jedná se sice o značně obecný popis, ale autor NETTE frameworku se touto cestou snaží zdůraznit, že je velice důležité vytvářet jednoduché a přehledné aplikace.

6.3 Stručný popis postupu při vytváření aplikací v NETTE frameworku

Následující text si neklade za cíl seznamovat případného zájemce o tuto technologii formou tutoriálu nebo podrobného popisu jednotlivých funkcí. Toto by bylo pochopitelně nad rámec této práce. Na internetu a zejména na [www stránkách NETTE – Quickstart](http://www.stránkách.NETTE-Quickstart)⁶ jsou zpracovány rady a postupy při tvorbě aplikací v NETTE. Vzhledem k tomu, že se jedná o novou technologii, tak se řada těchto návodů ještě postupně vyvíjí a rozšiřuje.

Následující text slouží spíše jako teoretický popis postupu tvorby aplikací v NETTE, aby si případní zájemci mohli udělat lepší představu o této technologii. Konkrétní informace už je pak třeba hledat na příslušných [www stránkách](http://www.stránkách).

6.3.1 Instalace frameworku

Pro tvorbu aplikace ve frameworku NETTE je třeba nejdříve stáhnout ze stránek NETTE – Download⁷ aktuální verzi frameworku. V adresáři `tools/Skeleton` se nachází užitečná pomůcka pro vytvoření adresářové struktury aplikace. Je zde vytvořena adresářová struktura, která je doporučena pro tvorbu aplikací.

```
PrvniAplikace/  
+--app/  
|   +--log/  
|   +--models/  
|   |   + ...  
|   +--presenters/  
|   |   + ...  
|   +--temp/  
|   +--templates/  
|   |   + ...  
|   +--bootstrap.php  
|   +--config.ini  
+--document_root/  
|   +--css/  
|   +--js/  
|   +--index.php  
|   + ...  
+--libs/  
+--tests/  
+--readme.txt
```

⁶ <http://doc.nettephp.com/cs/quickstart>

⁷ <http://nettephp.com/cs/download>

Do této kostry aplikace do adresáře `PrvniAplikace/libs` dále zkopírujeme samotný adresář Nette, který obsahuje celý framework.

Po provedení těchto kroků můžeme otestovat, zda vše funguje. Na adrese `http://localhost/PrvniAplikace/document_root/` by se měla zobrazit uvítací stránka NETTE frameworku.

Dále je potřeba nainstalovat knihovnu pro přístup k databázi. Jedná se o knihovnu dibi, která je ke stažení na stránce DiBi - Download⁸. Tuto knihovnu rozbalíme do adresáře `PrvniAplikace/libs` obdobně jako samotný framework.

Posledním krokem z instalační části je příprava jQuery. Zde stejně jako v předchozích případech z adresy JQuery – Download⁹ stáhneme příslušný instalační soubor a rozbalíme jej opět do adresáře `PrvniAplikace/libs`, ve kterém tedy budou adresáře Nette, dibi a jquery.

6.3.2 Struktura aplikace

Zde budou blíže popsány jednotlivé adresáře z vytvořené struktury aplikace pro seznámení s jejich významem a funkcí.

V adresáři `app` se nachází veškerá aplikační logika. Rozdělení logicky vyplývá z použitého modelu MVC. Nachází se zde tedy adresář `presenters`, kde jsou uloženy presentery aplikace, adresář `models` s modely pro aplikaci, adresář `templates` se šablonami a adresář `temp`, kde jsou ukládána dočasná data.

Důležitou součástí je soubor `bootstrap.php`, což je zaváděcí soubor celého frameworku, který inicializuje prostředí aplikace. Při vytváření adresářové struktury podle výše uvedeného postupu získáme i tento soubor v jeho výchozím nastavení.

Dále se zde nachází soubor `config.ini`, kde je uložena konfigurace samotné aplikace.

Dalším adresářem je `document_root`, kde se nacházejí veřejně dostupné soubory aplikace, jako jsou `css` soubory s kaskádovými styly, `js` soubory s potřebnými `javascripty`, případně obrázky s grafikou aplikace.

Důležité je zmínit, že tato adresářová struktura je pouze doporučeným postupem. NETTE framework nabízí značnou volnost při tvorbě aplikací a tuto strukturu nijak nevynucuje. Dle mého názoru však nemá smysl měnit tuto fungující a prověřenou strukturu aplikace.

6.3.3 Presenter

Framework NETTE používá návrhový vzor MVP – Model – View - Presenter, které je do značné míry podobný návrhovému vzoru MVC – Model – View - Controller.

První částí aplikace je tedy presenter. Ten je zároveň řídicí vrstvou aplikace. Každý presenter může provádět několik činností, které jsou označovány jako akce.

⁸ <http://dibiphp.com/cs/download>

⁹ <http://jquery.com/>

Presentery tvoří hierarchii v rámci aplikace. Tuto hierarchii je však třeba navrhovat pečlivě, aby se zamezilo situaci, kdy by byla hierarchie navržena špatně a jeden presenter by obsahoval mnoho akcí nebo naopak kdy by akce v rámci logické skupiny byly rozděleny.

Při návrhu hierarchie presenterů by každá akce měla obecně reagovat na jeden druh požadavků od klienta a tyto akce by měly být rozděleny podle souvislostí do jednotlivých logických skupin. V případě, že více presenterů má stejnou akci, můžeme ji použít pro jejich abstraktního předka. Při návrhu presenterů bychom měli dodržovat to, aby byl nestavový, měl akce související pouze s jednou činností. Zde můžeme postupovat podle jednotlivých akcí ve scénářích Use Case. Dále by každý presenter měl být abstrakt nebo final.

Jak již bylo uvedeno výše, framework NETTE obsahuje základní (doporučenou) adresářovou strukturu, v rámci které jsou již připravené kromě konfiguračních souborů a příslušných adresářů i některé základní presentery. Ty se nacházejí v adresáři app/presenters. Zajímavý je například BasePresenter.php, ve kterém lze nastavit společné chování pro všechny ostatní presentery. Dále se ve skeletonu aplikace nacházejí presentery Error, Homepage a Login.

Po vytvoření presenteru je potřeba implementovat jednotlivé akce presenteru. Bližší ukázka presenteru a dalších vrstev aplikace bude následovat vždy v další kapitole.

Ukázka implementace presenteru:

```
<?php
use Nette\Application\AppForm,
    Nette\Forms\Form;

class UcitelVypisZnamkyPresenter extends BasePresenter
{
    public function renderDefault()
    {
        $stridy = new UcitelVypisZnamky;
        $this->template->abs = $stridy->findAll();
    }

    public function renderView($id = 0)
    {
        $pocetchlapcu = 0;
        $pocetdivek = 0;
        $trida = new UcitelVypisZnamky;
        $this->template->tr = $trida->findZaci($id);

        $stat = $trida->findZaci($id);
        foreach ($stat as $key => $val) {
            if (($val->zak_pohlavi)==0){
                $pocetchlapcu++;
            }

            if (($val->zak_pohlavi)==1){
                $pocetdivek++;
            }
        }
        $this->template->pocetchlapcu = $pocetchlapcu;
        $this->template->pocetdivek = $pocetdivek;
    }
}
```

Obrázek č. 13: Ukázka presenteru

6.3.4 Šablona

Na rozdíl od návrhového vzoru MVC, kde se písmenem V označuje View – pohled, se ve frameworku NETTE používá označení šablona.

Tato šablona má na starosti veškeré zobrazování výstupů aplikace uživateli.

Šablona je tvořena souborem .phtml s názvem podle příslušné akce. Tento soubor je potřeba umístit do adresáře `app/templates/nazev_presenteru`.

Soubory šablon obsahují HTML kód. Pro každou akci jednotlivého presenteru je potřeba vytvořit jednu šablonu.

Framework NETTE umožňuje použití dvoufázového referování šablon. Jedná se o obdobu Masterpage používanou v jazyce C# v MS Visual Studiu.

Šablona je tedy rozdělena na dvě části. První část je společná pro více stránek. Může se tedy jednat o hlavičku, patičku nebo další části stránek, které se opakují ve stejné podobě na více místech.

Tyto soubory jsou uloženy přímo v adresáři `app/templates/` a jejich název začíná znakem `@`. Tento znak je umístěn v názvu proto, aby nebylo možné zaměnit šablony layoutu (společné pro více stránek) s jednotlivými šablonami pro konkrétní akce.

Ukázka implementace – šablona layoutu

Tato šablona slouží jako základ pro zobrazení celé aplikace. Jak jde vidět v uvedené ukázce, nachází se zde hlavička HTML stránky a HTML kód pro zobrazení společných částí stránky. Části, které se mění a jsou definovány v jiných šablonách, jsou propojeny kódem {include #content}. Na základě definování proměnné \$role je vloženo příslušné menu pro přihlášeného uživatele.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="cs">
  <head>
    <meta name="description" content="Nette Framework example" />
    <title>{block #title}{/block}</title>
    <link rel="stylesheet" type="text/css" media="screen" href="{basePath}/css/site.css" />
    <meta http-equiv="Content-Type" content="application/xhtml+xml; charset=utf-8" />
    <meta http-equiv="Content-language" content="cs" />
  </head>

  <body>
    <div id = "hlavicka" >
    </div>
    <div id = "obsah">
      <div id="vlevo">
        {if $role=='zak'}
        <p class="obdobi">{obdobi_nazev}</p>
        <a href="{plink ZakVypisAbsence:default}">Výpis absence</a><br />
        <a class="vlevo" href="{docroot}/zk/rozvrhy/zobraz_rozvrh_trida.php">Rozvrh</a><br />
        {/if}
        {if $role=='ucitel'}
        <p class="obdobi">{obdobi_nazev}</p>
        <a href="{plink UcitelVypisZnamky:default}">Statistiky známek</a><br />
        <a class="vlevo" href="{docroot}/zk/spolecne/zobraz_zamestnanec.php?zamestnanec_id={$uzivatel_id}">
        Osobní profil</a><br />
      </div>
      <div id = "pravy">
        {foreach $flashes as $flash}
        <div class="flash {$flash->type}">
        {$flash->message}
        </div>
        {/foreach}
        {include #content}
      </div>
    </div>
    <div id = "paticka">
    </div>
  </div>
</body>
</html>
```

Obrázek č. 14: Ukázka šablony layoutu

Ukázka implementace – šablona obsahu

V této šabloně se definuje část stránky označená jako #content pro zobrazení výpisu třídní knihy.

```
{block #content}

<h1>{block #title}Statistiky známek{/block}</h1>

Medián známek pro celou třídu je: {$median}<br />
Průměr známek pro celou třídu je: {$prumer}<br /><br />

Celkový počet zadaných známek: {$pocetznamek}<br /><br />

Počet žáků je: {$pocetzaku}<br />
Z toho počet chlapců: {$pocetchlapcu}, procent: {$procentachlapcu}<br />
Z toho počet dívek: {$pocetdivek}, procent: {$procentadivek}<br /><br />

<strong>Statistiky pro jednotlivé žáky</strong><br />
<table border = "1">
<tr>
<th>Žák</th>
<th></th>
</tr>
</tr>
{foreach $abs as $vypis}
<tr>
<td>{$vypis->zak_prijmeni} {$vypis->zak_jmeno}</td>
<td><a href="{link znamkyzaka, $vypis->zak_id}">Zobraz</a></td>
</tr>
{/foreach}
</table>
```

Obrázek č. 15: Ukázka šablony obsahu

6.3.5 Model

Model je vrstva aplikace, která získává data (obecně z databáze) a předává je presenteru a šabloně. Ve frameworku NETTE se obvykle používá pro přístup k databázi databázová vrstva dibi, která je blíže popsána v [8]. Zatím stačí jen zmínit, že je potřeba se k databázi připojit, což lze realizovat buď v konfiguračním souboru bootstrap.php nebo v BaseModelu. Konkrétní ukázky budou uvedeny dále v textu.

Pro vytvoření modelu je potřeba vytvořit dvě třídy ve složce app/models. Toto řešení vychází z návrhového vzoru Active Record [9], kde jedna třída představuje záznam z databáze a druhá umožňuje vrácení daných záznamů.

Ukázka připojení k DB

Připojení k databázi probíhá ve dvou krocích. V prvním kroku se v souboru config.ini nastaví přístupové údaje k databázi, které mohou zároveň obsahovat údaje pro testovací i produkční databázi a v dalším kroku se přímo vytváří připojení tak, jak je zřejmé z následujících ukázek kódu.

```
[production < common]
; common database connection
database.driver = mysql
database.database = zsupaplickycz1
database.charset = utf8
database.lazy = TRUE
database.host = localhost
database.username = root
database.password =

[development < production]
; database options in development mode
database.profiler = TRUE
database.username = produkcní_login
database.password = produkcní_heslo
```

Obrázek č. 16: Ukázka nastavení připojení k databázi

Ukázka zdrojového kódu modelu

```
<?php
use Nette\Object,
    Nette\Environment;

class UcitelVypisZnamky extends Object
{
    private $connection;

    public static function initialize()
    {
        dibi::connect(Environment::getConfig('database'));
    }

    public function __construct()
    {
        $this->connection = dibi::getConnection();
    }

    public function findZaci($id)
    {
        $rok_id = $_SESSION['akt_rok'];
        $trida_id = $_SESSION['trida'];

        return dibi::fetchAll('SELECT * FROM [zak]
            INNER JOIN [zak_trida]
                ON [zak_trida.zak_id]=[zak.zak_id]
            INNER JOIN [aktualni_trida]
                ON [zak_trida.trida_id]=[aktualni_trida.trida_id]

            WHERE
                [zak_trida.trida_id]=' . $trida_id . '
                AND [aktualni_trida.rok_id]=' . $rok_id . '
            ');
    }
}
```

Obrázek č. 17: Ukázka modelu

6.4 Popis protokolu HTTPS

Jak již bylo zmíněno v bakalářské práci, v případě, že jsou v IS ukládána a hlavně přenášena citlivá data, jako jsou rodná čísla a osobní údaje, je třeba klást velký důraz na bezpečnost těchto dat. Vzhledem k tomu, že data v samotné databázi jsou relativně dobře zabezpečena, nejslabším článkem je přenos dat mezi klientem - uživatelem (v tomto případě představovaném internetovým prohlížečem) a serverem. Tato komunikace standardně probíhá prostřednictvím HTTP protokolu.

6.4.1 Popis protokolu HTTP

Hypertext transfer protokol je protokol, který slouží pro komunikaci mezi internetovým prohlížečem a serverem a obecně pro přenos informací v prostředí internetu.

Protokol funguje na principu zasílání a přijímání textových zpráv mezi klientem a serverem a potvrzování zpráv pomocí příslušných kódů.

Ukázka HTTP komunikace podle [4]:

Požadavek klienta (v tomto případě internetový prohlížeč Mozilla):

```
GET /wiki/Wikipedie HTTP/1.1
Host: cs.wikipedia.org
User-Agent: Mozilla/5.0 Gecko/20040803 Firefox/0.9.3
Accept-Charset: UTF-8,*
```

Odpověď serveru:

```
HTTP/1.0 200 OK
Date: Fri, 15 Oct 2004 08:20:25 GMT
Server: Apache/1.3.29 (Unix) PHP/4.3.8
X-Powered-By: PHP/4.3.8
Vary: Accept-Encoding, Cookie
Cache-Control: private, s-maxage=0, max-age=0, must-revalidate
Content-Language: cs
Content-Type: text/html; charset=utf-8
```

Z výše uvedené ukázky je zřejmé, že komunikace probíhá v textové formě bez jakéhokoli šifrování. Je tedy možné, a v praxi toto není ani příliš obtížné, odposlouchávat takovouto komunikaci v síti. Proto vznikl HTTPS protokol, který je rozšířením HTTP protokolu, jak je popsáno dále.

6.4.2 Popis protokolu HTTPS

HTTPS protokol rozšiřuje možnosti protokolu HTTP. Umožňuje zabezpečit spojení před odposloucháváním (což řeší problém z výše uvedené ukázky komunikace) a dále umožňuje ověřovat identitu jednotlivých stran a zabránit podvržení dat třetí stranou. Zatímco HTTP protokol funguje na portu 80, protokol HTTPS používá port 443.

Popis použitého šifrování

Protokol HTTPS používá asymetrické šifrování. To znamená, že jsou použity odlišné klíče pro šifrování a dešifrování dat. Pro šifrování se používá veřejný klíč a pro dešifrování klíč soukromý. Detailně je tato problematika popsána například zde [5]. Detailní popis principů šifrování není náplní této diplomové práce.

6.4.3 Certifikáty a certifikační autority

Jak již bylo zmíněno, pro šifrování přenosu mezi klientem a serverem v protokolu HTTPS potřebujeme šifrovací klíče. Z důvodů, které budou dále uvedeny, je potřeba, aby byl šifrovací klíč důvěryhodný. Je tedy třeba, aby byl digitálně podepsaný důvěryhodnou certifikační autoritou. Zde je však třeba rozlišovat certifikační autoritu od obecně uznávané důvěryhodné certifikační autority.

Certifikační autoritou může být například běžný správce webhostingu. Dokonce i jakýkoli uživatel může být sám sobě certifikační autoritou. Mezi mezinárodní uznávané certifikační autority však patří jen několik málo subjektů. Například se jedná o VeriSign, THAWTE, RapidSSL. U ostatních certifikačních autorit může být problém s tím, že se uživatelům zobrazí varování v jejich internetovém prohlížeči. I když toto nemá na funkčnost a bezpečnost certifikátů žádný zásadní vliv, běžného laického uživatele toto varování zajisté odradí od návštěvy stránek. Vzhledem k tomu, že IS je již od počátku vyvíjen se snahou o co nejjednodušší použití, tak varování o nedůvěryhodném certifikátu představuje z tohoto pohledu značný problém.

Pro provoz serveru s vlastním certifikátem od uznávané certifikační autority je potřeba jednak webhosting s vlastní IP adresou a dále samotný certifikát. Zde je z praktického hlediska problém hlavně ve finanční stránce, což je obecně problém jakéhokoli zabezpečení. Zde jsou však náklady v porovnání s cenami standardního webhostingu značně vysoké.

Zmíněný provoz serveru s vlastní IP adresou vyjde podle [6] na 350kč/měsíc. Vzhledem k tomu, že běžná cena za standardní webhosting se pohybuje kolem 100kč/měsíc, je tato částka značně vysoká.

K ceně za provoz serveru je potřeba také připočítat samotný certifikát. Ten stojí v základní variantě od 1000kč/rok podle [6] a [7]. Zde se však nejedná o certifikát uznávané certifikační autority, ale o certifikáty vydané v tomto případě provozovateli uvedených webhostingů. V případě, že je potřeba certifikát od uznávané certifikační autority, například od firmy VeriSign, je základní cena certifikátu 10 900kč/rok podle [11]. Zde se tedy dostáváme přibližně na cenu 15 000kč za celkový provoz webhostingu s protokolem HTTPS, případně cca 5000kč za provoz bez certifikátu od uznávané certifikační autority. Tato částka je pro školu pochopitelně značně vysoká. Zvláště v poměru k ceně současného webhostingu, který stojí cca 1500kč/rok. Zde bohužel narážíme na typický problém obecně jakéhokoli zabezpečení, kterým je cena takového řešení. Do budoucna je tedy jen na finančních možnostech školy, zda se rozhodne používat tuto vyšší úroveň zabezpečení.

Technické informace ohledně nastavení vlastního webového serveru pro provoz protokolu HTTPS jsou uvedeny v příloze B.

6.4.4 Shrnutí provozu protokolu HTTPS

Z výše uvedených podmínek pro provoz IS s použitím protokolu HTTPS je zřejmé, že tento protokol je sice znatelně bezpečnější než základní protokol HTTP, ale je třeba počítat se znatelně vyššími náklady na jeho provoz. Toto je však pro základní školy poměrně zásadní problém. Pokud bychom počítali s běžnými náklady na webhosting s databází potřebný pro provoz www stránek a IS ve výši cca 1500kč, tak náklady na HTTPS protokol jsou znatelně vyšší. Podle ceníku společnosti Forpsi [6] je třeba počítat s náklady ve výši 350kč/měsíc za provoz serveru a minimálně cca 1000kč za základní certifikát. Ten však není příliš vhodný z výše uvedených důvodů. V případě certifikátu od obecně uznávané certifikační autority by bylo potřeba k nákladům na provoz serveru přidat minimálně 2000kč za certifikát.

Jako alternativa k řešení od komerční firmy poskytující webhosting by šlo zvážit pořízení vlastního serveru. Tímto řešením by šlo ušetřit náklady za webhosting a provoz serveru na protokolu HTTPS. Ušetřená částka by se pohybovala okolo 5000kč/rok. Při započítání nákladů na pořízení serveru, internetovou konektivitu a správu serveru by se však náklady pohybovaly pravděpodobně ještě na vyšší úrovni než standardní komerční řešení.

6.5 Instalace

Pro provoz IS je třeba mít připravený server s uvedenými HW a SW požadavky a s příslušnými uživatelskými účty.

Samotná instalace se provádí nastavením databáze a překopírováním zdrojových souborů. Pro vytvoření databáze je třeba použít soubor `database.sql`, který obsahuje skripty pro vytvoření celé struktury databáze a její naplnění potřebnými počátečními daty. Jako první uživatel pro spuštění IS slouží uživatel s loginem `admin` a heslem `zakladniskola`.

Dále je třeba zkopírovat adresář `/zk/` do požadovaného umístění na serveru.

Posledním krokem je konfigurace veškerého nastavení v souboru `config.php`. Tento konfigurační soubor je popsán v samostatné kapitole.

6.6 Popis jednotného konfiguračního souboru

Jak již bylo výše uvedeno, pro jednodušší konfiguraci IS při počátečním spuštění je třeba sjednotit údaje potřebné pro celkovou konfiguraci do jednoho souboru. Původně bylo zamýšleno vytvořit nový konfigurační soubor `config.ini`, ve kterém by byly tyto informace uloženy. Při další analýze se však ukázalo, že pro zjednodušení bude stačit použít již existující soubor `connect.php`, ve kterém jsou nyní uloženy jen přístupové údaje k databázi.

Do tohoto souboru byly přidány definice konstant pro nastavení údajů, které bylo třeba měnit při spuštění IS. Zde se vycházelo ze zkušeností se spuštěním IS na ZŠ Karla Dvořáčka. Konfigurační soubor tedy obsahuje kromě již používaných údajů pro přístup k databázi nastavení textu pro titulek stránky, text v patičce, kontakt na administrátora, cestu ke složce se soubory IS, rozepsané pořadí a délku trvání vyučovacích hodin a nastavení vah známek.

Tyto konfigurační údaje lze snadno změnit jak při počáteční instalaci, tak během provozu IS.

6.7 Použití HTTPS protokolu

Protože přenos dat pomocí HTTPS protokolu probíhá v aplikační vrstvě TCP/IP protokolu, je provoz aplikací s využitím HTTPS protokolu do značné míry nezávislý na samotné PHP technologii. Z důvodu bezpečnosti je pouze třeba v každé stránce kontrolovat, zda je pro komunikaci mezi klientem a serverem používán protokol HTTPS. Vzhledem k tomu, že část IS implementovaná v rámci bakalářské práce je řešena mírně odlišným způsobem, je třeba použít dvě metody pro ověřování HTTPS komunikace. V následujícím kódu je znázorněna kontrola pro část implementovanou v rámci bakalářské práce.

```
<?PHP
// port pro SSL komunikaci
$SSL_Port=443;
// v pripade, ze komunikace neprobiha pres protokol HTTPS,
automaticky presmerujeme komunikaci na tento protokol
if ($SERVER_PORT!=$SSL_Port) {
    if (header("Location: https://$HTTP_HOST$SCRIPT_NAME"));
```



```

else
// v pripade, ze spojeni neni ani po pokusu o presmerovani uspesne,
vypiseme o tom informaci
    echo "Nelze navázat komunikaci přes HTTPS protokol";
    exit;
}
// zde v testovacim priklade vypiseme pro kontrolu informaci o
pouzivani SSL protokolu
?>

```

Používáte SSL protokol.

V případě částí implementovaných v NETTE frameworku se ověření může provádět v metodě `isSecured()` třídy `Nette\Web\HttpRequest` nebo při nastavování routovacích pravidel v souboru `index.php`, kde se pro ověření, zda se jedná o HTTPS komunikaci, používá parametr `SimpleRouter::SECURED`.

6.8 Popis implementace nového typu uživatele – sekretářka

V následujících kapitolách bude popsána implementace částí IS, které zasahují do již implementovaných a používaných modulů. Nové funkce jsou popsány samostatně v návrhu implementace a dalších příslušných kapitolách.

Tento IS byl od počátku navrhován s cílem vytvořit obecný IS, který by navíc měl umožňovat co nejjednodušší rozšiřování.

Po více než roce provozu IS měla ZŠ Karla Dvořáčka požadavek na přidání dalšího typu uživatele. Toto je relativně velký zásah do struktury a funkcí IS. Vzhledem k tomu, že se však s rozšiřitelností IS počítalo od počátečního návrhu, nebylo přidání nového typu uživatele příliš problematické.

Informace o jednotlivých typech uživatelů jsou obsaženy na několika místech:

- databázová tabulka uživatel – atribut `zamestnanec_role`,
- formulář pro přidávání nového uživatele pro administrátora,
- soubor `login.php`,
- soubory s hlavičkou a menu pro daný typ uživatele,
- soubor `kontrola_prihlaseni.php`,
- soubor `menu.php`.

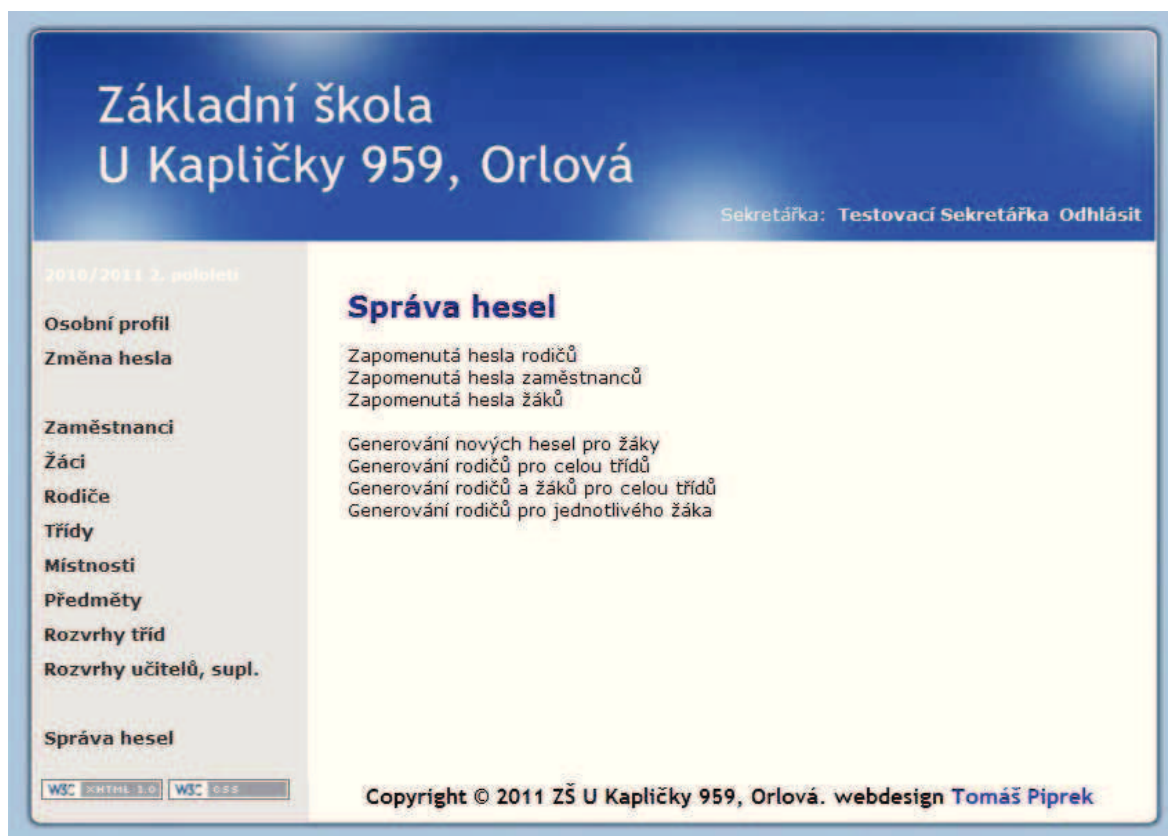
Pro přidání je tedy potřeba upravit výše uvedené výskyty odkazů na daný typ uživatele.

- V databázové tabulce `zamestnanec` není potřeba atribut `zamestnanec_role` nijak upravovat. Pouze se k označení „a“ pro administrátora a „u“ pro učitele začne používat označení „s“ pro sekretářku.
- Aby bylo možné nový typ uživatele používat, je potřeba umožnit administrátorům IS vytvářet nové uživatele. Zde stačí přidat v souboru `add_zamestnanec.php` k možnostem u

role uživatele k typu učitel a administrátor nový typ uživatele sekretářka. Také je potřeba nastavit, aby se daný typ uživatele vkládal do databáze s atributem `zaměstnanec_role` nastaveným na hodnotu „s“.

- Při přihlašování do aplikace se v souboru `login.php` kontroluje správnost zadaného loginu a hesla a podle toho, o jaký typ uživatele se jedná, je uživatel přesměrován na příslušnou stránku. Zde je tedy třeba přidat nový typ uživatele a nastavit přesměrování po přihlášení na stránku `sekretarka.php`.
- Pro každý typ uživatele se provádí kontrola přihlašování. Proto je třeba připravit soubor `kontrola_prihlaseni.php` a upravit jej pro nový typ uživatele.
- Posledním krokem je příprava funkcí, které budou dostupné novému typu uživatele. IS je navržen tak, aby bylo možno používat jednotlivé funkce pro více typů uživatelů. V tomto případě stačí vytvořit uživatelské menu, kde budou obsaženy odkazy na jednotlivé funkce (výpis žáků, výpis tříd, přehled rozvrhů).

Podle potřeby je možné implementovat další funkce, které zatím nebyly implementovány pro žádný typ uživatele.



Obrázek č. 18: Ukázka rozhraní pro uživatele Sekretářka

6.9 Popis implementace importů žáků

Jak již bylo uvedeno v zadání, operace vkládání většího množství žáků byly s použitím základních vstupních formulářů časově příliš náročné. Proto bylo rozhodnuto o vytvoření funkce pro hromadné importy žáků a následné automatické generování jejich hesel a rodičů.

Protože základní školy mají ze zákona povinnost evidovat údaje o žácích v elektronické podobě a pro tuto evidenci používají program DM Evidence, bude třeba použít výstupy z tohoto programu pro import dat do IS.

Z exportních formátů, které podporuje program DM Evidence, byl vybrán formát CSV. V tomto programu však nejsou evidovány všechny údaje o žácích. Pro provoz IS je zejména důležitý login a pohlaví žáků (potřebné pro seřazování známek). Po diskuzi s ICT metodiky byl zvolen postup, kdy se vyexportovaný CSV soubor z programu DM Evidence doplní v Excelu o chybějící hodnoty a následně se použije pro import v IS pomocí funkce Import dat.

Po importu žáků lze dále pokračovat vygenerováním jejich hesel a rodičů. V praxi tedy administrátorovi IS stačí vytvořit seznam žáků pro import a rodiče (loginy i hesla) a hesla žáků jsou již vygenerovány automaticky a stačí je pouze předat příslušným žákům nebo rodičům.

6.10 Popis implementace generování PDF výpisů známek

Protože se v nejbližší době plánuje zrušení papírových žákovských knížek, je třeba připravit výpisy známek s ohledem na jednoduché generování a efektivní tisknutí. Z tohoto důvodu byl pro generování výpisů známek použit formát pdf. Tento formát je vhodnější pro bezproblémové tisknutí a také vhodnější pro archivaci. Vzhledem k tomu, že při počtu žáků okolo 500 a například při dvoutýdenní frekvenci výpisů je jednotlivých listů s výpisy značné množství, bylo při implementaci této funkce testováno několik rozvržení stránek s výpisy, velikostí písma a celkového vzhledu stránky tak, aby bylo zajištěno co nejefektivnější využití jednotlivých listů papíru. Ukázka výpisu známek je uvedena v příloze E.

6.10.1 Technické řešení implementace

Pro implementaci samotného generování pdf souborů byla vybrána knihovna mPDF [16]. Mezi její hlavní výhody patří jednoduché použití, podpora kódování UTF-8 a tím pádem podpora českých znaků a licence GNU GPL v2 [17], která umožňuje volné použití.

```
<?php
require("mpdf.php");
$mpdf = new mPDF('utf-8', 'A4');
$mpdf->useOnlyCoreFonts = true;
$mpdf->SetDisplayMode('fullpage');
$mpdf->SetAutoFont(0);

// CSS soubor
$stylesheet = file_get_contents('test.css');
// vstupni soubor v HTML
$html = file_get_contents('test.html');

$mpdf->WriteHTML($stylesheet,1);
$mpdf->WriteHTML($html,2);

$name = "vypis-znamek.pdf";
$mpdf->Output($name,"D"); // download
?>
```

Obrázek č. 19: Ukázka použití knihovny mPDF pro generování pdf souboru s výpisem známek

6.10.2 Ukázka struktury výpisu známek

Datum	Předmět	Učivo	Známka	Učitel

7 Testování

Během vývoje nových funkcí IS v rámci diplomové práce bylo postupováno obdobně jako u bakalářské práce. Nejprve byla připravena první testovací verze, která byla dále konzultována s klíčovými uživateli v jednotlivých školách. Tímto způsobem bylo možno implementovat všechny funkce přesně podle požadavků škol.

Dále byly průběžně sledovány podněty od všech uživatelů IS a podle jejich priorit byly dále buď opravovány nalezené chyby, nebo přidávány nové funkce.

V rámci testování však nestačilo testovat jen nové funkce. Vzhledem k tomu, že některé nové funkce, jako je vytvoření uživatelské role Sekretářka, hromadné generování uživatelů a podobně zasahovaly do současné struktury IS, bylo třeba průběžně testovat i již implementované funkce a integraci nových částí se staršími.

7.1 Testovací data

Testování IS probíhalo převážně na ZŠ U Kapličky a částečně na ZŠ Karla Dvořáčka. Během provozu IS se v databázích obou škol uložilo značné množství dat. Na začátku testování byly vytvořeny kopie databází a ty byly použity pro testování. Bylo tak zajištěno dostatečné množství dat pro testování a nebylo třeba vytvářet testovací data ručně. Toto testování s databází s větším množstvím dat by mělo být efektivnější, vzhledem k tomu, že při testování IS v rámci bakalářské práce se ukázalo, že některé chyby se objevily až při reálném testování uživateli z důvodu, že ve skutečném provozu bylo vkládáno mnohem větší množství dat než při testování.

Větší testovací databáze také umožnila lepší otestování rychlosti IS, která se pochopitelně projeví až s vzrůstajícím počtem údajů v databázi.

Tyto databáze obsahují interní údaje škol, a proto byly použity pouze při testování v rámci daných škol, a v této diplomové práci nejsou tyto údaje obsaženy.

Pro testování a spuštění IS mimo uvedené školy je připravena počáteční testovací databáze se základními údaji, jako jsou základní uživatelé. Použití této databáze je podrobněji popsáno v kapitole implementace.

Základní škola U Kapličky 959, Orlová

Učitel: **Franta Novák** [Odhlásit](#)

2010/2011 2. pololetí

[Třídní kniha](#)

[Testy](#)

[PDF výpisy známek](#)

[Statistiky známek](#)

[Osobní profil](#)

[Editovat osobní údaje](#)

[Změna hesla](#)

[Sdělení žák](#)

[Sdělení třída](#)

[Vzkazy od rodičů](#)

[Známkování](#)

[Přehled známek](#)

["E-learning"](#)

[Zaměstnanci](#)

[Žáci](#)

[Třídy](#)

[Rozvrhy tříd](#)

[Rozvrhy učitelů, supl.](#)

W3C XHTML 1.0

W3C CSS

PDF výpisy známek

Přehled vytvořených výpisů

Datum od	Datum do	Datum vytvoření	Žák	
2011-04-13	2011-04-21	2011-04-02	Vojtěch Přechek	Smazat
2011-04-06	2011-04-13	2011-04-02	Hana Bernatíková	Smazat
2011-04-01	2011-04-15	2011-04-18	Jaroslav Kornecký	Smazat

Vytvoření nového výpisu

Žák:

Datum od:

Datum do:

duben 2011

po	út	st	čt	pá	so	ne
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

Copyright © 2011 ZŠ U Kapličky 959, Orlová. webdesign [Tomáš Pipek](#)

Obrázek č. 20: Ukázka vytváření výpisů známek

Základní škola
U Kapličky 959, Orlová

Učitel: Petr Novák Odhlásit

2010/2011 2. pololetí

Třídní kniha
Testy
PDF výpisy známek
Statistiky známek
Osobní profil
Editovat osobní údaje
Změna hesla

Sdělení žák
Sdělení třída
Vzkazy od rodičů
Známkování

Třídní kniha

Pondělí

Matematika	Zlomky	Absence
Dějepis	První světová válka	Absence
Přírodopis	Prvoci	Absence
Český jazyk	Diktát	Absence
Angličtina	Gramatika	Absence
Chemie	Kyseliny	Absence
Domácnost	Vaření	Absence

Uložit

Obrázek č. 21: Ukázka třídní knihy

7.2 Zaškolení

U zaškolení se osvědčila strategie, kdy byli definováni klíčoví uživatelé, v tomto případě ICT metodici školy, se kterými byly řešeny všechny záležitosti. Tito klíčoví uživatelé měli na začátku připravenou testovací verzi IS, na které si vyzkoušeli veškeré potřebné úkony. Testování probíhalo také průběžně během implementace nových funkcí, kdy klíčoví uživatelé postupně testovali jednotlivé verze funkcí IS během jejich vývoje. Po důkladném otestování a seznámení se s IS už tito klíčoví uživatelé zaškolili ostatní zaměstnance školy a dále sami řešili běžné provozní záležitosti týkající se provozu IS.

8 Analýza dat

V dnešní době se již z informačních systémů stává standardní vybavení všech firem. Dobře to lze pozorovat na zavádění IS v prostředí základních škol. Na začátku tvorby IS v rámci bakalářské práce uvedené školy neměly žádné zkušenosti s provozem IS. Během doby implementace a používání IS se však jeho používání stalo standardní součástí provozu školy. Je zřejmé, že když se moderní informační technologie již dostaly i do oblasti základních škol, ve kterých je obecně nedostatek finančních prostředků a nízké znalosti informačních technologií, tak v komerční sféře jsou již obdobné IT technologie samozřejmostí.

Obvykle se ale IS používá pouze pro běžné administrativní úkony. Po několika letech používání každého IS se v něm však začnou hromadit data, která většina společností nevyužívá.

Využitím těchto dat se zabývá rozsáhlá oblast business intelligence. Náplní této diplomové práce však není teoretický popis principů BI. Ty jsou blíže popsány například v [18] a [19].

V dalších kapitolách bude tedy uveden praktický postup při analýze dat získaných během provozu IS na ZŠ Karla Dvořáčka.

8.1 Popis datových skladů

Databáze běžných IS pracují na principu OLTP (Online Transaction Processing). Toto řešení je však vhodné jen pro samotný provoz IS, kde jsou nejčastější operace vkládání a vyhledávání a kde je důležité zamezení výskytu redundantních údajů a splnění 3. NF, jak je popsáno v [18] a [19]. Pro provoz datových skladů je naopak vhodnější princip zpracování dat OLAP (Online Analytic Processing), kde je výskyt redundantních údajů žádoucí z důvodu zrychlení práce s databází. Zdrojová data a záměrně vytvořené redundance se rozdělují na fakty a dimenze a dohromady vytvářejí multidimenzionální databázi, ze které vychází celý datový sklad. Toto je samozřejmě jen velmi zjednodušená charakteristika sloužící k popsání základních rozdílů. Zbytek teoretického popisu zde není uveden vzhledem k zaměření této části diplomové práce.

8.1.1 Dimenze a fakta

Vzhledem k tomu, že vyhledávání v DS (datovém skladu) probíhá na odlišném principu než vyhledávání v OLTP databázích, je třeba rozdělit atributy pro vytvoření multidimenzionální databáze na fakta a dimenze.

Dimenze jsou atributy, podle nichž je potřeba sumovat a agregovat jiné atributy. Typickým příkladem je dimenze podle času, učitele nebo předmětu.

Fakty představují údaje, které nás zajímají v tom smyslu, že je vypisujeme podle uvedených dimenzí. Typickým faktem bude například cena zboží nebo známka žáka pro naši databázi.

Kromě faktů a dimenzí je ještě třeba evidovat některé ostatní převážně popisné atributy, jako například jméno učitele, název předmětu nebo popis výsledku žáka.

8.1.2 Multidimenzionální model

V běžných databázích jsou data reprezentována ve formě tabulek, mezi nimiž jsou vytvořeny vazby pomocí klíčů. Každá tabulka obsahuje jednotlivé záznamy.

Pro datové sklady se používá naprosto odlišný model. Data jsou zde reprezentována multidimenzionálním modelem, který bývá označován jako multidimenzionální kostka. V tomto modelu jsou ukládány již předzpracované údaje, které jsou obvykle agregovány a sumarizovány podle předem zvolených parametrů a již záměrně obsahují redundantní údaje. V této multidimenzionální kostce tvoří jednotlivé dimenze její osy a fakty jsou tvořeny průniky těchto dimenzí.

8.1.3 Datový sklad

Výše popsaná multidimenzionální kostka je základním prvkem datového skladu. Datový sklad je tedy strukturovaným úložištěm dat zaměřených na určitou společnost s cílem podporovat rozhodování obecně pro vyšší management dané společnosti.

Přesněji je zde uvedena definice podle [19].

Datový sklad je integrovaná, subjektivě orientovaná, stálá a časově rozlišitelná sbírka dat, uspořádaná pro podporu potřeb managementu.

8.1.4 Tvorba datového skladu

Stejně jako při implementaci běžných programů nebo informačních systémů se lze i při tvorbě datových skladů řídit různými standardními postupy.

Pro datové sklady se jedná o metody velkého třesku a metodu přírůstkovou, která se dále dělí na postup „zdola nahoru“ a „shora dolů“. Tyto metody jsou blíže popsány v [21]. Pro tvorbu datového skladu bude v našem případě použita metoda velkého třesku, která bude blíže popsána v kapitole o vlastním řešení datového skladu.

8.1.5 ETL

Datový sklad typicky obsahuje vstupní data z různorodých informačních zdrojů od klasických OLTP databází s různou strukturou až po stará archivní data v různých zastaralých formátech nebo dokonce získána z papírové evidence. Cílem procesu ETL je získat (Extract), transformovat (Transform) a načíst (Load) zdrojová data do datového skladu a tímto naplnit multidimenzionální kostku. Složitost těchto operací roste s různorodostí zdrojů dat. V případě našeho datového skladu budou tyto procesy jednodušší vzhledem k tomu, že budou použita data z jedné OLTP databáze. Konkrétní postup bude uveden dále.

8.2 Popis vlastního datového skladu

V následujících kapitolách bude popsán praktický postup při analýze a implementaci datového skladu. Bude tedy popsán výběr zdrojových dat a jejich potřebné úpravy, navrhnutí a vytvoření datového skladu, přenos dat do datového skladu a jednotlivé výsledky analýzy dat.

Zdrojové kódy týkající se analýzy dat jsou uloženy na CD v adresáři /zdrojove_kody/analyza_dat/ ve dvou projektech. Jedná se o první projekt pro integraci (PIP009 – Integration project) a druhý projekt (PIP009 – Analysis project) pro analýzy dat.

8.2.1 Popis zdrojových dat

Pro analýzu dat budou použity údaje z databáze ZŠ Karla Dvořáčka, které byly vloženy do IS jeho uživateli v období školních let 2009/2010 a 2010/2011. Za tuto dobu se nahromadilo značné množství dat. Pro analýzu dat z IS základní školy budou pochopitelně nejzajímavější údaje o známkách žáků. V tabulce *vysledky*, ve které jsou jednotlivé známky žáků ukládány, se nachází přes 100 tisíc záznamů. Pro analýzu bude tedy použita uvedená tabulka *vysledek* a příslušné tabulky, které zde mají obsaženy cizí klíče, což jsou tabulky *zak*, *predmet*, *zaměstnanec*, *skolni rok* a *aktualni_trida*.

8.2.2 Popis zdrojových tabulek a atributů

Zak (zak_id, zak_login, zak_heslo, zak_jmeno, zak_prijmeni, zak_ulice, zak_mesto, zak_psc, zak_e-mail, zak_datum_narozeni, zak_rodne_cislo, zak_okres_narozeni, zak_misto_narozeni, zak_narodnost, zak_statni_prislusnost, zak_pojistovna, zak_druzina, zak_jidelna, zak_teldomu, zak_mobil, zak_smazan)

Zamestnanec (zamestnanec_id, zamestnanec_login, zamestnanec_heslo, zamestnanec_titul_pred, zamestnanec_jmeno, zamestnanec_prijmeni, zamestnanec_titul_z, zamestnanec_kabinet, zamestnanec_email, zamestnanec_aprobace, zamestnanec_telefon, zamestnanec_role, zamestnanec_smazan)

Aktualni_trida(trida_id, trida_rocnik, trida_oznaceni, mistnost_id, zamestnanec_id, rok_id, trida_poznamka, trida_smazana)

Predmet (predmet_id, predmet_zkratka, predmet_nazev, predmet_poznamka, predmet_smazan)

Skolni_rok (rok_id, rok_rok, rok_zacatek_pol, rok_konec_pol, rok_aktualni)

Vysledek (vysledek_id, zak_id, predmet_id, vysledek_datum, vysledek_popis, vysledek_znamka, vysledek_vaha, zamestnanec_id, vysledek_precteno, rok_id, vysledek_smazano)

8.2.3 Rozdělení na dimenze a fakta

Pro implementaci datového skladu je třeba rozdělit údaje ze zdrojových tabulek na požadovaná fakta a dimenze. V tomto případě bude tabulka faktů obsahovat výsledky žáků, které jsou hlavním bodem zájmu pro analýzu. Dimenze pak obsahují atributy týkající se času, tříd, zaměstnanců a podobných atributů, podle kterých se budou sestavovat přehledy.

Fakta:

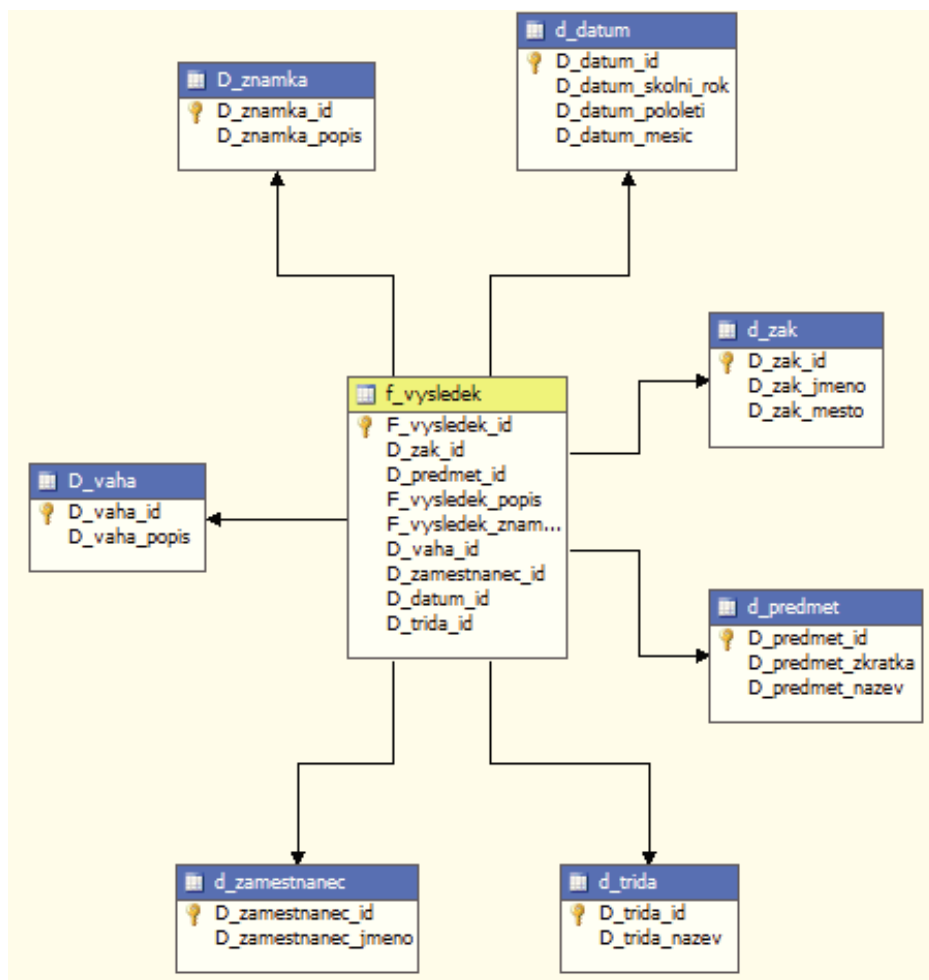
F_Vysledky (F_vysledek_id, D_zamestnanec_id, D_zak_id, D_predmet_id, F_vysledek_popis, D_vysledek_znamka, D_vysledek_vaha)

Dimenze:

D_Predmet (D_predmet_id, D_predmet_nazev, D_predmet_zkratka)
D_Zamestnanec (D_zamestnanec_id, D_zamestnanec_jmeno)
D_Zak (D_zak_id, D_zak_jmeno, D_zak_mesto)
D_Datum (D_datum_id, D_skolni_rok, D_pololeti, D_mesic)
D_Trida (D_trida_id, D_trida_nazev)
D_Vaha (D_vaha_id, D_vaha_popis)
D_Znamka (D_znamka_id, D_znamka_popis)

8.2.4 Schéma datového skladu

Na obrázku 22 je zobrazeno schéma datového skladu vytvořené podle výše uvedeného rozdělení na fakta a dimenze. V tomto případě se jedná o hvězdicové schéma, které má ve svém středu tabulky F_vysledek.



Obrázek č. 22: schéma datového skladu

8.2.5 Nepoužité atributy

Z výše uvedeného rozdělení atributů na fakta a dimenze je zřejmé, že nebyly použity zejména popisné atributy, jejichž evidence má smysl jen pro běžné administrativní potřeby školy. Seznam použitých a nepoužitých atributů je zřejmý ze schématu datového skladu a rozpisu faktů a dimenzí uvedeného výše.

8.2.6 Popis programového vybavení

Pro realizaci analýzy dat bylo vybráno řešení od firmy Microsoft. Potřebné programy se skládají z databáze MSSQL, ve které jsou uložena vstupní data, MSSQL Management studia, které slouží pro správu databáze, Visual Studio, které je použito pro implementaci ETL procesů, vytvoření datového skladu a dalších analýz a MS Excelu, který je použitý pro sestavování reportů a grafů z výstupních dat. Přesné verze použitých nástrojů jsou uvedeny níže.

Přehledy výstupních dat:	MS Excel 2007
Implementace:	MS Visual Studio 2008 Version 9.0.30729.1 SP
Správa databáze:	Microsoft SQL Server Management Studio 10.0.1600.22
Databáze:	Microsoft SQL Server Developer Edition 10.0.1600.22

8.2.7 Integrace dat - ETL

V rámci tvorby datového skladu bývají obvykle ETL procesy náročnou částí implementace. V tomto případě jsou však použita data z jedné databáze, a proto odpadá nutnost sjednocování dat z různých zdrojů. Pro další použití dat tedy stačí odstranit údaje, které nemá smysl evidovat v datovém skladu. Obecně se jedná o smazané položky v databázi a popisné atributy.

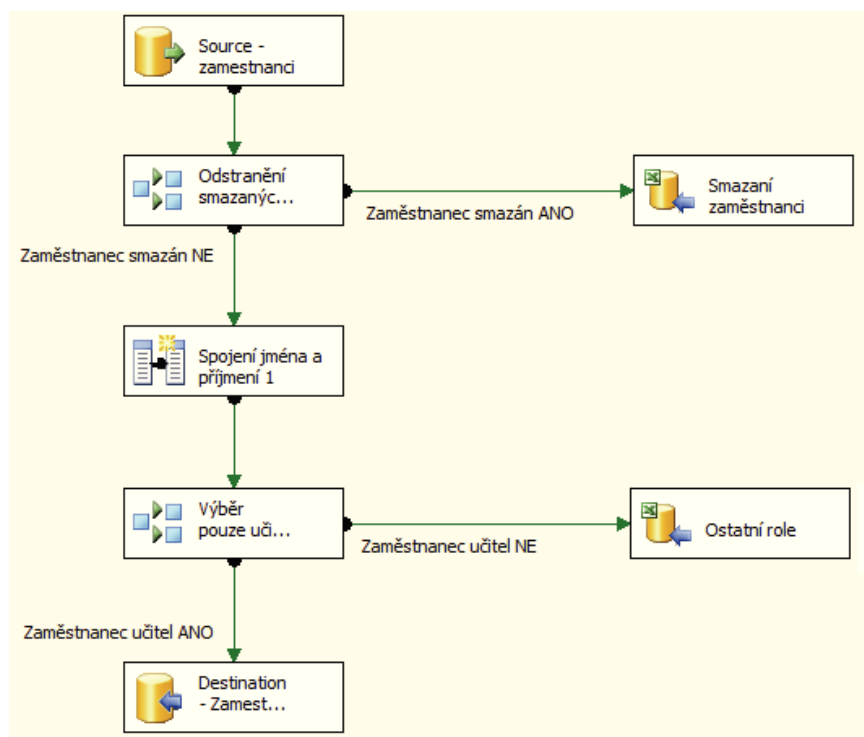
Dále budou uvedeny jednotlivé tabulky použité pro datový sklad a příslušné kroky, které byly implementovány pro zavedení dat do datového skladu. V praxi tyto kroky odpovídají jednotlivým částem v SSIS balíčcích pro jednotlivé tabulky.

- **Zak** – podle atributu zak_smazan budou odstraněny záznamy pro smazané žáky. Dále byly odstraněny atributy zak_ulice, zak_e-mail, zak_datum_narozeni, zak_rodne_cislo, zak_okres_narozeni, zak_misto_narozeni, zak_narodnost, zak_statni_prislusnost, zak_pojistovna, zak_druzina, zak_jidelna, zak_teldomu, zak_mobil, zak_smazan. Tyto atributy mají jen popisný charakter a vzhledem k počátečnímu záměru použití IS jsou často nevyplněny z důvodu evidence údajů v jiném systému. Atributy zak_jmeno a zak_prijmeni budou sloučeny do jednoho atributu zak_jmeno.
- **Zamestnanec** – podle atributu zamestnanec_smazan budou odstraněny záznamy pro smazané zaměstnance. Podle atributu zamestnanec_role budou vybráni pouze zaměstnanci podle role učitel, která je jako jediná relevantní pro evidenci známek. Budou odstraněny

popisné atributy `zamestnanec_heslo`, `zamestnanec_titul_pred`, `zamestnanec_titul_z`, `zamestnanec_kabinet`, `zamestnanec_email`, `zamestnanec_aprobace`, `zamestnanec_telefon`, `zamestnanec_role`, `zamestnanec_smazan`, které mají jen popisnou roli.

Atributy `zamestnanec_jmeno` a `zamestnanec_prijmeni` budou sloučeny do jednoho atributu `zamestnanec_jmeno`.

- **Aktualni_trida** - podle atributu `trida_smazana` budou odstraněny smazané třídy a dále budou odstraněny atributy `mistnost_id`, `trida_poznamka`, `trida_smazana`, `rok_id`, `zamestnanec_id`.
- **Predmet** - podle atributu `predmet_smazan` budou odstraněny smazané předměty a dále budou odstraněny popisné atributy `predmet_poznamka` a `predmet_smazan`.
- **Skolni_rok** – budou odstraněny atributy `rok_zacatek_pol`, `rok_konec_pol`, `rok_aktualni`. Atribut `rok_rok` bude rozdělen na 2 nové atributy `datum_skolni_rok`, `datum_pololeti`. Hodnota původního atributu `rok_rok` „2009/2010 1. pololetí“ bude tedy rozdělena na „2009/2010“ a „1. pololetí“.
- **Vysledek** - podle atributu `vysledek_smazano` budou odstraněny smazané výsledky a dále budou odstraněny atributy `vysledek_smazano`, `rozvrh_id`, `vysledek_precteno`, které dále nejsou třeba.
Z atributu `vysledek_datum` bude extrahováno číslo měsíce, které bude vloženo do dimenzionální tabulky `D_datum`. Jedinečné hodnoty atributů `vysledek_znamka` a `vysledek_vaha` budou vloženy do tabulek `D_znamka` a `D_vaha`.
- **Datum** – údaje v této tabulce budou naplněny ve výše uvedených krocích z tabulek `vysledek` a `skolni_rok`.
- **Vaha** – tabulka bude obsahovat hodnoty `Id` vložené při zpracování tabulky `vysledek` a jednotlivé popisy pro hodnoty váhy známek, které je třeba vložit ručně podle hodnot procentuální váhy známek používaných ve škole.
- **Znamka** - tabulka bude obsahovat hodnoty `Id` vložené při zpracování tabulky `vysledek` a jednotlivé popisy pro hodnoty známek, které je třeba vložit ručně podle požadovaného popisu známek (výborně, chvalitebně, atd.).



Obrázek č. 23: Ukázka implementace ETL procesu pro tabulku zaměstnanec v prostředí Business Intelligence Development Studio

Na výše uvedeném obrázku č. 23 je znázorněn ETL proces pro záznamy z tabulky *Zamestnanec*, kde se provádí odstranění smazaných zaměstnanců, výběr zaměstnanců s rolí učitel podle atributu *zaměstnanec_role* a sjednocení jména z atributů *zaměstnanec_jmeno* a *zaměstnanec_prijmeni*.

8.2.8 Přehled statistických údajů

Jak již bylo uvedeno, pro analýzu dat jsou použity údaje z dvouletého provozu IS. V následující tabulce jsou uvedeny počty záznamů z jednotlivých tabulek pro získání přehledu o množství analyzovaných dat. Uvedené počty jsou již uvedeny pro odstranění údajů v rámci ETL procesů.

Tabulka	Počet záznamů
Vysledek	95 624
Zak	466
Zamestnanec	36
Predmet	29
Trida	37

8.2.9 Vytvoření OLAP kostky

Po provedení výše uvedených kroků jsou pro datový sklad připraveny tabulky dimenzí a faktů, které jsou v během ETL procesů naplněny požadovanými daty. Dalším krokem tvorby datového skladu je vytvoření OLAP kostky (Cube). Tato kostka představuje multidimenzionální reprezentaci údajů ze vstupních tabulek.

Pro její vytvoření je třeba v prostředí BIDS definovat zdroj dat, vytvořit nad ním pohled, definovat dimenze a specifikovat požadované vstupní údaje (dimenze, fakta a jejich atributy) pro vytvoření kostky.

8.3 OLAP analýza a reporty

Pro vytváření výstupních sestav z údajů z vytvořené kostky byl použit MS Excel, který umožňuje připojení ke kostce a prohlížení dat pomocí kontingenční tabulky. Na následujícím obrázku je znázorněna výstupní sestava představující počty vložených známek rozdělené podle období, žáků a předmětů.

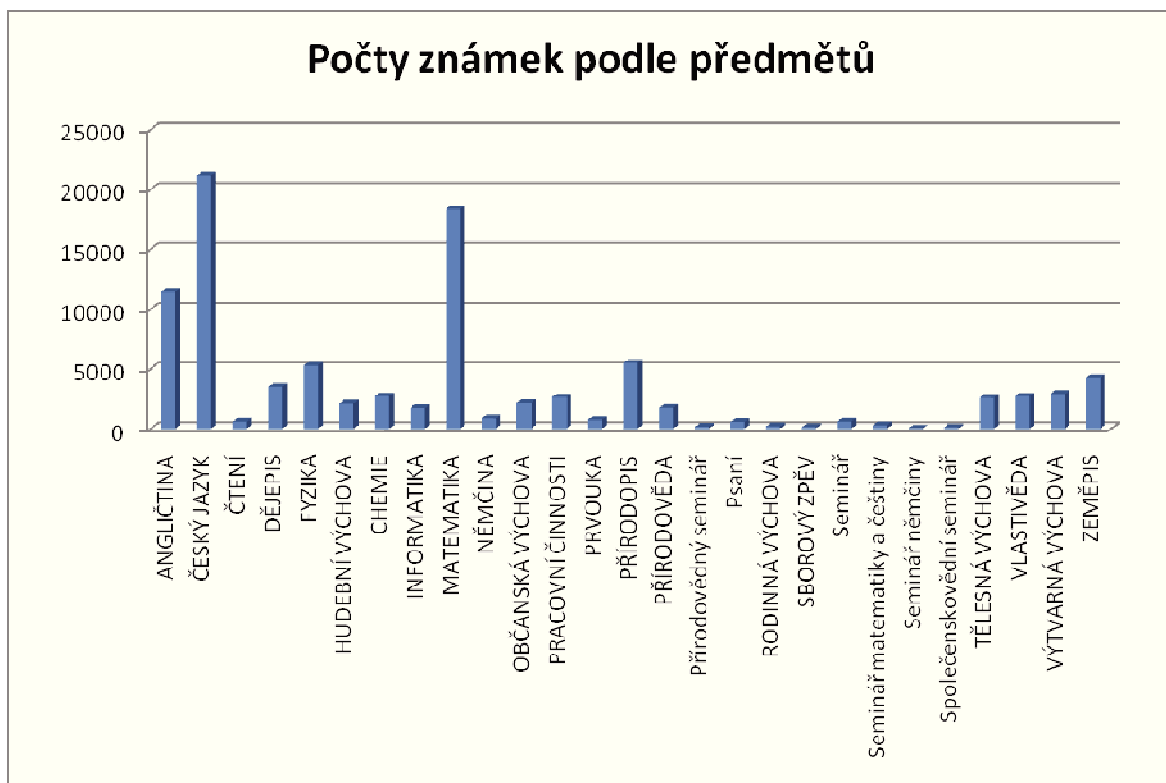
Celkový počet známek		Popisky sloupců		Celkem 2009/2010	Celkový součet
2009/2010					
Popisky řádků		1.	2.		
Adam Cieslar		92	86	178	178
- ANGLIČTINA		7	4	11	11
Květoslava Kaziměřčiková		7	4	11	11
- ČESKÝ JAZYK		17	19	36	36
Hynek Tešnar		17	19	36	36
- FYZIKA		13	5	18	18
Alice Nogová		13	5	18	18
- HUDEBNÍ VÝCHOVA		2		2	2
Jan Raszka		2		2	2
- CHEMIE		8	7	15	15
Věra Bořková		8	7	15	15
+ INFORMATIKA		4	6	10	10
+ MATEMATIKA		16	14	30	30
+ PRACOVNÍ ČINNOSTI		1		1	1
+ PŘÍRODOPIS		10	13	23	23
+ RODINNÁ VÝCHOVA			4	4	4
+ TĚLESNÁ VÝCHOVA		2	4	6	6
+ VÝTVARNÁ VÝCHOVA		3	2	5	5
+ ZEMĚPIS		9	8	17	17
Adam Feber			74	74	74
- ANGLIČTINA			13	13	13
Květoslava Kaziměřčiková			13	13	13
- ČESKÝ JAZYK			28	28	28
Danuše Pytlíková			28	28	28
+ MATEMATIKA			13	13	13
- PŘÍRODOVĚDA			5	5	5
Danuše Pytlíková			5	5	5
+ TĚLESNÁ VÝCHOVA			1	1	1
- VLASTIVĚDA			12	12	12
Danuše Pytlíková			12	12	12
+ VÝTVARNÁ VÝCHOVA			2	2	2
Celkový součet		92	160	252	252

Obrázek č. 24: Ukázka výstupní sestavy v prostředí MS Excel

V následujícím textu budou uvedeny některé výsledky analýzy dat. Tyto výstupy byly navrženy na základě konzultací s vedením školy a představují tak reálně využitelné informace.

8.3.1 Počty známek podle předmětů

Na následujícím grafu je uveden vztah počtu vložených známek a jednotlivých předmětů. Je zde na první pohled zřejmé, že nejvíce známek je vloženo pro předměty angličtina, český jazyk a matematika. Jedná se o nejdůležitější předměty, které jsou vyžadovány při přijímacích zkouškách na střední školy, a proto na ně vyučující kladou největší důraz. Větší počet známek také do jisté míry reprezentuje náročnost předmětů na přípravu ze strany žáků.



Graf 1. počty známek podle předmětů

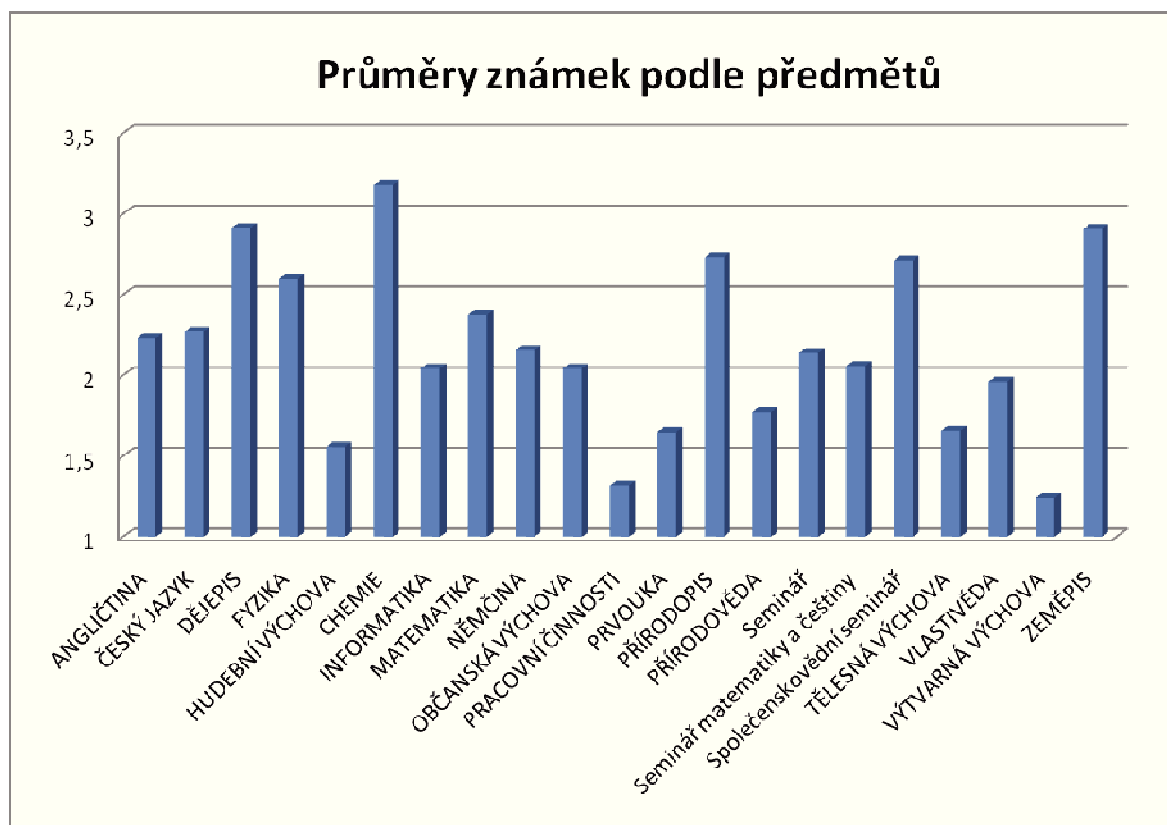
8.3.2 Průměrné známky podle předmětů

Na následujícím grafu jsou znázorněny celkové průměrné známky za jednotlivé předměty. V obecném měřítku by tyto průměry měly představovat náročnost předmětů.

Z výsledků je zřejmé, že předměty dějepis, fyzika, chemie, přírodopis, společenskovědní seminář a zeměpis jsou hodnoceny nejhoršími známkami. Toto odpovídá obecnému názoru na náročnost technických předmětů. Naopak předměty hudební výchova, pracovní činnosti a výtvarná výchova jsou hodnoceny převážně známkou 1.

Z grafu je také zřejmá odlišná úroveň známkování na prvním a druhém stupni školy. To vyplývá z průměrných známek pro předměty přírodopis a přírodověda. Oba předměty, jak název napovídá, mají podobnou náplň výuky, ale přírodopis je vyučován na prvním stupni a přírodověda na druhém stupni. I přes podobnou učební osnovu se průměr znatelně liší. Průměrná známka

z přírodovědy je 1,8, zatímco průměrná známka z přírodopisu je 2,7. Je zde zřejmé, že na druhém stupni se kladou přísnější požadavky na žáky.



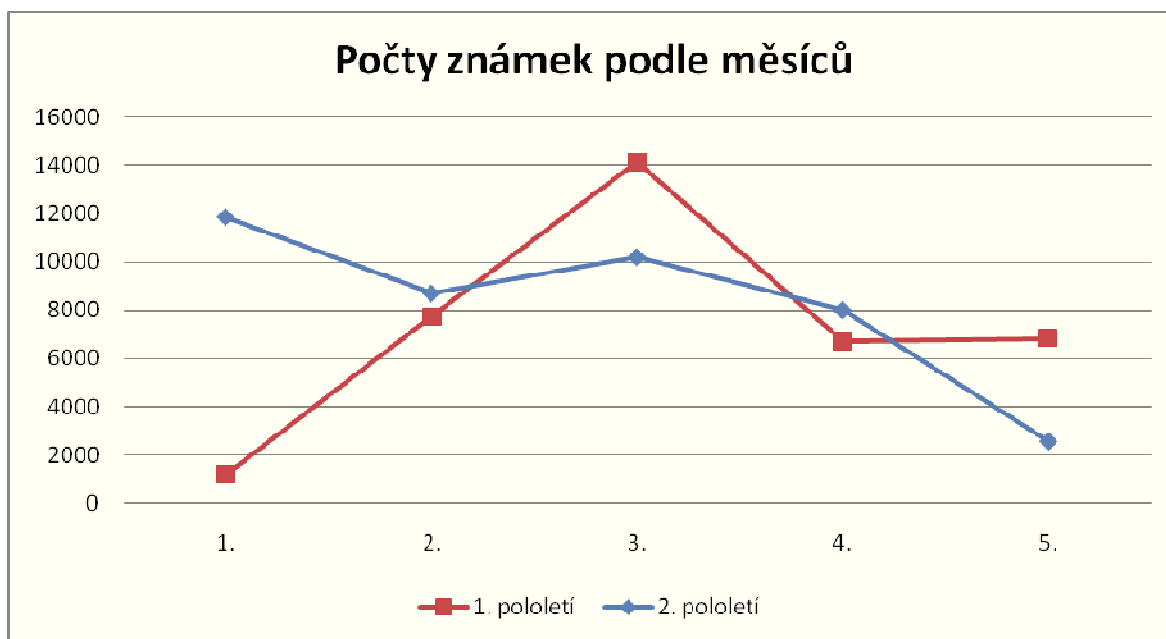
Graf 2. průměry známek podle předmětů

8.3.3 Počty známek podle měsíců

V následujícím grafu je uvedeno množství známek zadávané v jednotlivých měsících. V přeneseném smyslu toto číslo může znázorňovat náročnost výuky v jednotlivých měsících, vzhledem k tomu, že počet známek představuje množství testů, domácích úkolů a dalších hodnocení žáků, z kterého lze odvodit náročnost výuky v čase.

Graf znázorňuje počty známek v 1. a 2. pololetí roku podle příslušných měsíců. 1. měsíc prvního pololetí je září a 5. měsíc je leden a obdobně pro druhé pololetí.

Z grafu lze vyčíst, že na začátku roku se zadává menší množství známek a výuka tak probíhá pomalejším tempem. Nejvíce známek se zadá ve 3. měsíci prvního pololetí, což je listopad, kde se vyhodnocuje čtvrtletní prospěch. V prvním měsíci druhého pololetí se již zadává více známek než v prvním měsíci prvního pololetí. Ke konci školního roku se opět počet zadaných známek snižuje.

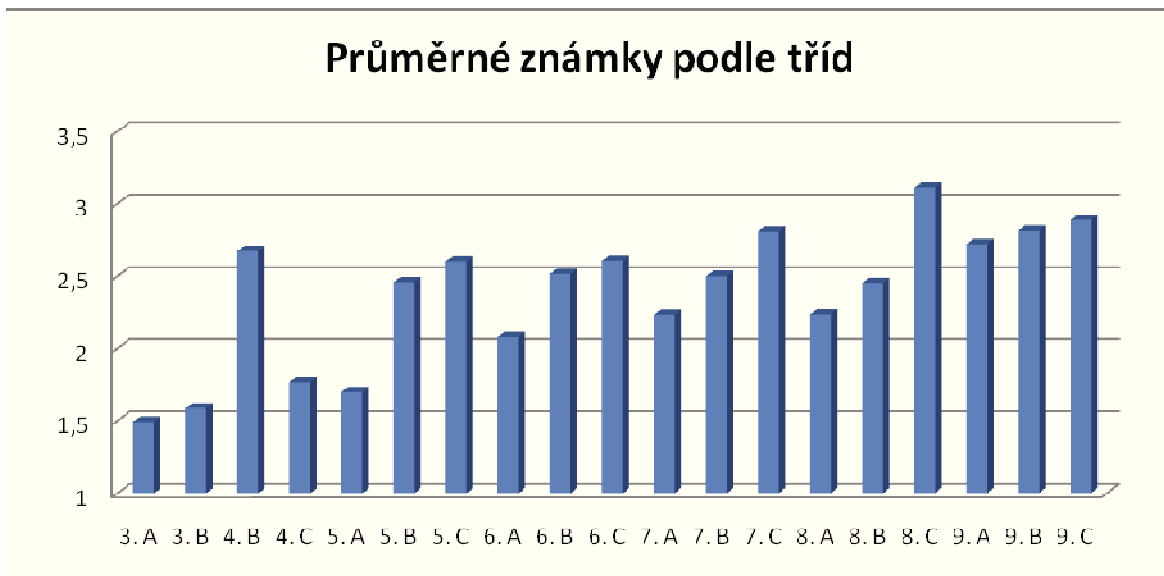


Graf 3. počty známek podle měsíců

8.3.4 Průměry známek podle tříd

Na následujícím grafu jsou uvedeny celkové průměry za jednotlivé třídy. Z grafu je zřejmé, že se ve vyšších třídách (s rostoucím věkem žáků) zvyšuje i průměrný prospěch. Zde lze spekulovat o tom, zda se žáci s rostoucím věkem méně věnují školním povinnostem nebo s rostoucím věkem učitelé zvyšují nároky na žáky.

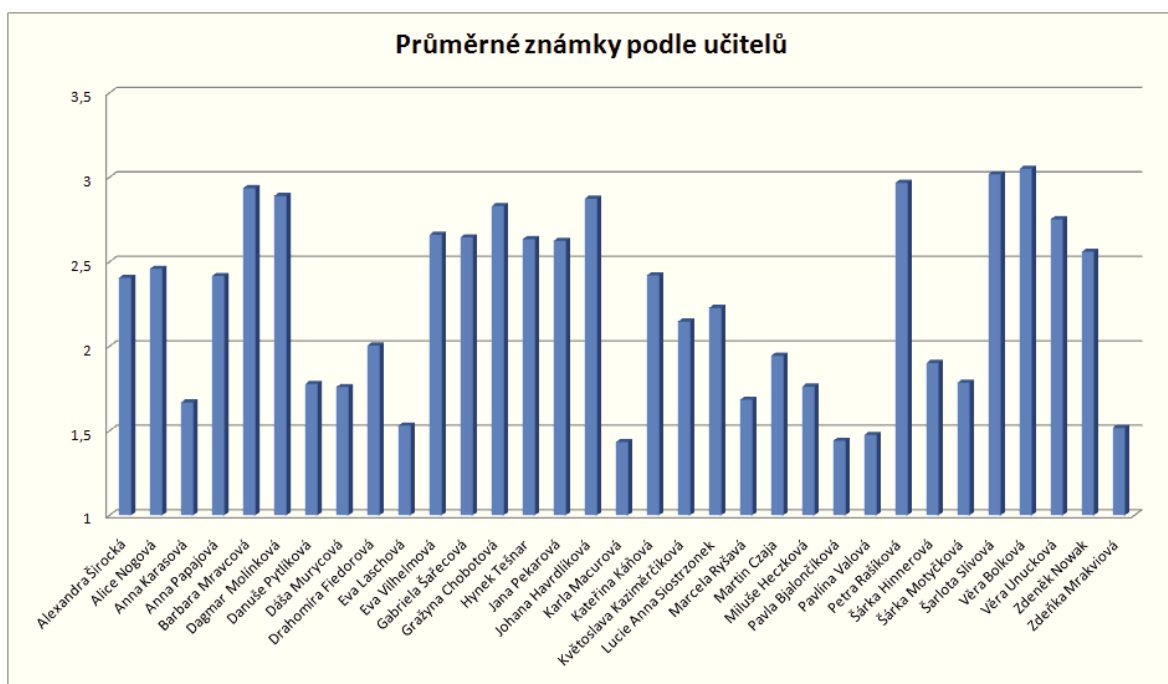
Další zajímavou informací z grafu je odlišný průměr v rámci jednoho ročníků. Kromě čtvrtého ročníku se tento průměr vždy liší podle označení třídy a to tak, že třída „A“ má nejnižší průměr a třída „C“ má nejvyšší průměr.



Graf 4. průměrné známky podle tříd

8.3.5 Průměry známek podle učitelů

Na dalším grafu jsou uvedeny průměry známek zadané jednotlivými učiteli bez ohledu na třídu a předměty. Opět se zde projevuje závislost mezi prvním a druhým stupněm základní školy a mezi prospěchem žáků. Učitelé s nižšími průměry známek, jako jsou Anna Karasová, Danuše Pytlíková, Dáša Murycová a podobně, jsou bez výjimky učitelé na prvním stupni. Z uvedeného grafu lze odvodit, že vyšší průměr zadávaných známek není způsoben větší přísností učitelů, ale obecně většími nároky na žáky druhého stupně, případně menší snahou těchto žáků.



Graf 5. průměry známek podle učitelů

8.4 Popis dataminingu

V předchozí kapitole byly uvedeny výsledky analýzy dat pomocí běžných statistických metod.

Cílem této kapitoly bude pokusit se navrhnout vhodné dataminingové metody pro pokročilejší analýzu zdrojových dat. Pro analýzu budou použita stejná zdrojová data jako v předcházející kapitole.

8.4.1 Popis programového vybavení pro datamining

Pro analýzu dat pomocí dataminingových metod bylo vybráno Business Intelligence Development Studio, stejně jako pro sestavování reportů v předchozí kapitole. Použití stejného prostředí umožňuje využít již připravenou databázi se vstupními daty případně rovnou analyzovat data z vytvořeného datového skladu.

8.4.2 Příprava dat pro datamining

Pro počáteční přípravu dat pro datový sklad bylo třeba provést celou řadu ETL operací. Díky tomu, že všechny záležitosti ohledně přípravy dat byly již provedeny, je příprava dat výrazně jednodušší. Pro přehlednost jednotlivých vstupů pro konkrétní dataminingové metody již stačí jen vytvořit pohledy nad údaji ze zdrojové databáze pro větší přehlednost a rychlost zpracování, nebo pracovat rovnou s celou databází.

8.4.3 Rozhodovací stromy

V následujících kapitolách budou vždy uvedeny jen použité metody, jejich nastavení a výsledky. Konkrétní popis principů jednotlivých metod a algoritmů je uveden v [18] a [19].

Jako první metoda pro analýzu dat byly vybrány rozhodovací stromy.

Pomocí rozhodovacích stromů se budeme snažit ověřit hypotézu, zda má na hodnotu známek vliv předmět, pro které jsou známky zadány, učitel, který známku vkládá (vyučuje daný předmět) a třída, pro kterou daný učitel vyučuje předmět.

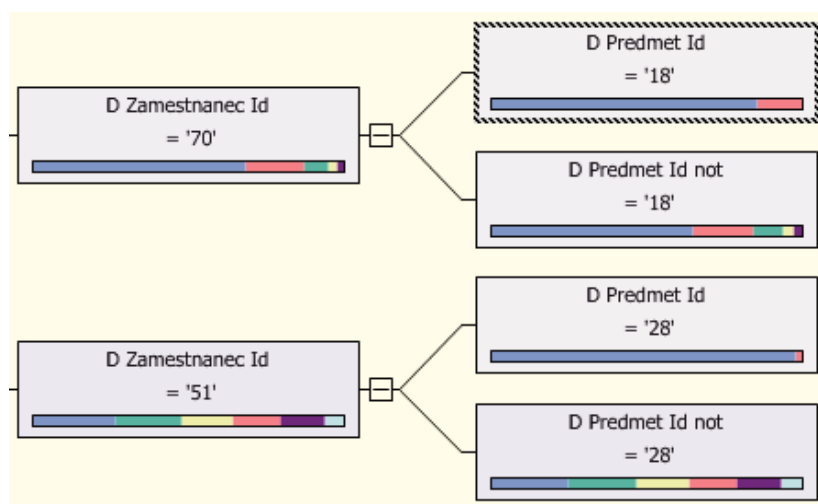
Pro tuto metodu byla použita tabulka F_vysledek. Vzhledem k tomu, že se jedná o tabulku, použitou jako tabulku faktů, jsou v ní již obsaženy všechny údaje potřebné pro tuto analýzu.

Nastavení algoritmu

Pro použití rozhodovacích stromů byl zvolen algoritmus Microsoft Decision Trees. Pro ten je třeba zvolit předpovídaný atribut, kterým je v tomto případě výsledek_znamka a vstupní atributy, kterými jsou predmet_id, trida_id a zaměstnanec_id. Dále byl algoritmus testován s různými hodnotami proměnných HoldoutMaxPercent a HoldoutSeed a pro jednotlivé hodnoty byly procházeny výsledky z výstupních rozhodovacích stromů.

Výsledek

Ve všech nastaveních algoritmu se vytvářely stromy s podobným rozložením, kde první úroveň představovala dělení podle učitelů a dále se výsledky dělily podle předmětů nebo tříd. Použitím metody rozhodovacích stromů se nepodařilo potvrdit hypotézu, že známky jsou obecně ovlivněny předmětem, třídou nebo učitelem. Ukázalo se, že i když žádná obecná závislost nebyla nalezena, u některých předmětů byly tyto závislosti nalezeny.



Obrázek č. 25: rozhodovací strom

Na výše uvedeném obrázku je část rozhodovacího stromu. Zaměstnance představují Dagmar Molínková (51) a Eva Laschová (70) a předměty Výtvarná výchova (18) a Seminář z matematiky a češtiny (28).

Z této části rozhodovacího stromu můžeme odvodit, že předměty Výtvarná výchova a Seminář z matematiky a češtiny mají vliv na výslednou známku tohoto předmětu.

V rámci tříd a zaměstnanců se nepodařilo nalézt žádnou závislost.

8.4.4 Shlukovací metody

Další testovanou metodou analýzy dat je shlukování. Tato metoda byla opět řešena pomocí BIDS, kde pro analýzu shlukování slouží algoritmus Microsoft Clustering Algorithm.

Cílem shlukování je nalézt ve vstupní množině dat charakteristické výrazné podmnožiny. Tento algoritmus je spíše běžné používat na množině dat, o které nemáme bližší informace o jednotlivých vztazích. V tomto případě, kdy jsou použita data ze základní školy, u kterých jsme dobře seznámeni s jejich účelem, strukturou a hodnotami dat, se dá odvodit předem předpokládaný výsledek shlukování. Je zřejmé, že data již jsou strukturována do podmnožin, které jsou představovány jednotlivými třídami, předměty nebo učiteli. Lze tedy očekávat, že výsledky analýzy pomocí shlukovacího algoritmu budou v jistých ohledech odpovídat těmto podmnožinám a celkové struktuře dat v IS.

Nastavení algoritmu

Pro shlukování je třeba zvolit klíčový atribut, který je představován atributem F_vysledek_id. Dalšími atributy byly D_datum_id, D_predmet_id, D_trida_id, D_zamestnanec_id a F_vysledek_znamka. S těmito atributy bylo experimentováno za použití různých nastavení počtu hledaných shluků a minimální podpory. Z atributů byly také vybírány různé podmnožiny (například D_zamestnanec_id a D_trida_id), z důvodu, že při některých kombinacích nastavení a použití všech uvedených atributů algoritmus nebylo možné spustit vzhledem k nedostatku operační paměti.

Výsledek

Během všech zkoušených kombinací nastavení algoritmu se projevila výše uvedená hypotéza, že u již strukturovaných dat nebude možné nalézt nové relevantní shluky dat. Shluky, které byly nalezeny, odpovídaly některým výsledkům z analýzy dat v předchozí kapitole. Byly nalezeny shluky pro předměty Matematika, Čeština a Angličtina, které odpovídají třem předmětům s nejvyšším počtem zadaných známek a dále například výrazný shluk pro známky „výborně“, které jsou nejčastější vloženou známkou. Ukázalo se, že shlukovací metoda nepřináší relevantní výsledky pro data, která již jsou logicky strukturována.

8.4.5 Asociace

Vzhledem k výsledkům předchozích analýz se i u asociací dá předpokládat obdobný výsledek, kde dataminingové metody pouze potvrdí obecně známá pravidla. I přesto bude vyzkoušeno co možná nejvíce nastavení algoritmu pro možné získání užitečných výsledků. Pro asociace se v BIDS používá Microsoft Association Rule algorithm.

Nastavení algoritmu

Jako vstupní data byly vybrány atributy D_datum_id, D_predmet_id, D_trida_id, D_zak_id, D_zamestnanec_id a F_vysledek_znamka obdobně jako u předchozí shlukovací metody. Při použití všech atributů najednou nebyl získán žádný smysluplný výsledek, vzhledem k tomu, že bez ohledu na nastavení jednotlivých parametrů vznikl vysoký počet asociačních pravidel, která však neměla žádnou vypovídající hodnotu. Dále tedy bylo experimentováno s různými podmnožinami uvedených atributů a s různým nastavením minimální a maximální podpory a spolehlivosti.

Výsledek

Opět se potvrdilo, že vzhledem k dobrému přehledu o struktuře dat jednotlivé výsledky asociací jen potvrdily již známá fakta. Například pro vstupní atributy D_predmet_id, D_trida_id a F_Vysledek_znamka byly získány následující údaje.

Probability	Importance	Rule
0,983	0,376	D Predmet Id = 19, D Trida Id = 16 -> F Vysledek Znamka = 1
0,957	0,361	D Predmet Id = 24, D Trida Id = 21 -> F Vysledek Znamka = 1
0,957	0,365	D Predmet Id = 25, D Trida Id = 8 -> F Vysledek Znamka = 1
0,941	0,356	D Predmet Id = 18, D Trida Id = 16 -> F Vysledek Znamka = 1
0,935	0,353	D Predmet Id = 24 -> F Vysledek Znamka = 1
0,929	0,350	D Predmet Id = 19, D Trida Id = 21 -> F Vysledek Znamka = 1

Obrázek č. 26: Výsledek hledání asociací

Z výsledku lze vyčíst, že v případě, že se jedná o předmět pracovní činnosti a třídu 6. A, tak jsou téměř všechny známky „výborné“. Obdobně je tomu pro předmět sborový zpěv a třídu 3. A. Ostatní asociace jsou jen jejich obdobou pro jiné třídy a předměty.

Při použití dalších podmnožin vstupních atributů jako například D_predmet_id, D_zamestnanec_id, D_trida_id se opět ve výsledcích objevovaly již známé vztahy typu zaměstnanec učí předmět v dané třídě. Toto jsou však vztahy známé všem uživatelům IS a nepřinesly tak žádné nové informace.

8.5 Zhodnocení analýzy dat

Při používání metod pro shlukování, rozhodovací stromy a asociace se ukázalo, že vzhledem k dobré znalosti vztahů jednotlivých entit v databázi se těmito metodami nepodařilo získat nové informace o těchto datech. Z výše uvedených výsledků je však zřejmé, že dataminingové metody potvrdily údaje získané v předchozí kapitole věnované reportům. V případě, že bychom měli k dispozici pro analýzu údaje, o kterých bychom neměli předem dostatek informací, získali bychom výše uvedenými metodami relevantnější výsledky.

9 Závěr

9.1 Základní škola U Kapličky

V rámci diplomové práce se dle mého názoru povedlo úspěšně navázat na IS vytvořený v rámci bakalářských prací [1] a [2]. Nové části byly implementovány pomocí frameworku NETTE, který zefektivnil průběh implementace a umožnil kvalitnější implementaci výsledných funkcí. Použití tohoto frameworku by do budoucna také mělo zjednodušit další údržbu a případné přidávání funkcí.

Dle plánu byly implementovány funkce pro evidenci třídní knihy, statistiky známek, vytváření testů a dále byly původní části IS rozšířeny o hromadné operace s uživateli a o roli sekretářky. Tyto funkce jsou připraveny a otestovány pro používání v běžném provozu školy.

9.1.1 Zhodnocení provozu IS

Od prvního spuštění testovací verze IS na ZŠ U Kapličky uplynuly již téměř tři roky. Toto je pro provoz IS poměrně dlouhá doba. Jak již bylo zmíněno v předchozích kapitolách, na začátku tvorby bakalářské práce se věnovalo značné úsilí samotnému vymezení požadavků na IS a vytvoření základů pro provoz mnoha IT technologií. Během reálného provozu IS vyvinutého v rámci bakalářské práce se povedlo implementovaný IS začlenit do každodenního provozu školy. V průběhu provozu IS se průběžně opravovaly nalezené chyby a shromažďovaly se požadavky na nové funkce, které vedly k jejich implementaci v rámci této diplomové práce.

Celkově hodnotím provoz IS jako úspěšný, vzhledem k tomu, že se jeho používání stalo standardní součástí školní agendy a s jeho provozem nebyly žádné vážnější problémy.

9.1.2 Ostatní IT

Jak již bylo zmíněno v předchozích kapitolách, používání informačních technologií při každodenní výuce se stalo standardem. Je však velice důležité, aby byly dostupné technologie doplněny také kvalitními výukovými materiály. Zpočátku si každý učitel připravoval tyto elektronické výukové materiály samostatně. Toto řešení sice usnadnilo učitelům běžnou práci, jako je například zdlouhavé vypisování na tabuli, ale z dlouhodobého hlediska není ideální. Jejich obsah se může u jednotlivých vyučujících lišit, může být obtížné například suplování za jiného učitele a také na tuto přípravu nelze čerpat finanční prostředky z rozpočtu školy na mzdy učitelů.

V rámci projektu EU peníze školám¹⁰ se podařilo získat dva milióny korun na rozvoj výše uvedených výukových materiálů. Díky tomuto projektu se nyní ve škole připravují elektronické výukové materiály ve formě šablon pro jednotlivé předměty. Ty umožní formalizovat tvorbu, strukturu a obsah výukových materiálů, čímž je zpřehlední, zlepší kvalitu a umožní je používat v rámci celé školy. Tato práce bude učitelům proplácena nad rámec jejich mzdy, což běžně není možné. Kromě výukových materiálů budou pořízeny učitelům notebooky a dataprojektory.

¹⁰ www.eupenizeskolam.cz

I přesto, že se školy obecně zaměřují na rozvoj používání elektronických materiálů při výuce, tak v rámci uvedeného projektu bude ve škole zřízena nová knihovna s klasickými papírovými knihami. Žáci tak budou i přes rozvoj informačních technologií stále pracovat s knihami a neztratí schopnost porozumět textu, což v poslední době řada učitelů označuje za velký problém.

9.2 Základní škola Karla Dvořáka

9.2.1 Zhodnocení provozu IS

Na ZŠ Karla Dvořáčka byl IS spuštěn o rok později než na ZŠ U Kapličky. Jeho spouštění a následný provoz byl již jednodušší. Náročnější však byl současný provoz IS na dvou školách zejména vzhledem k nárůstu administrativních úkonů spojených převážně s vytvářením nových tříd, nových žáků a rozvrhů na začátku školního roku. Běžné administrativní úkony jako přidávání jednotlivých žáků, generování zapomenutých hesel a ostatní běžné záležitosti byly průběžně řešeny ICT metodikem školy, panem Czajou.

Požadavky na změny a evidence chyb v IS byly souběžně řešeny na obou školách. Jednotlivé požadavky tak mohly být důkladněji analyzovány vzhledem k většímu množství informací od uživatelů používajících IS.

Nové funkce byly průběžně konzultovány s ICT metodikem a vedením školy.

I přes počáteční problémy s konfigurací domény školy byl celkový provoz IS bez závažnějších problémů. Před spuštěním IS naléhali rodiče na vedení školy kvůli urychlení jeho spuštění. Během dalšího provozu se však ukázalo, že z dlouhodobého hlediska rodiče sledují výsledky žáků v delších časových intervalech, ale více se zajímají o výsledky žáků zveřejňované na www stránkách školy, které však přímo nesouvisí s jejich studijními výsledky. Celkově je však znatelný výrazný nárůst zájmu o informace o žácích v elektronické podobě.

9.2.2 Analýza dat

V průběhu provozu IS na ZŠ Karla Dvořáčka se v databázi nahromadilo značné množství dat. Vzhledem k tomu, že při vyhodnocování výsledků se na základních školách často sestavují různé přehledy a statistiky známek, zajímalo se vedení školy o pokročilejší metody analýzy těchto dat.

Výsledky analýzy dat uvedené v kapitole Analýza dat byly prezentovány vedení školy a poskytly tak cenné informace o datech vložených do IS za dobu jeho provozu a hlavně seznámily vedení školy s novými možnostmi práce s daty.

Kladně byly hodnoceny zejména přehledy průměrných známek srovnávané podle tříd a učitelů. Dále byly pro vedení školy zajímavé grafy množství známek zadávaných v průběhu jednotlivých měsíců, ze kterých vedení školy získalo nové informace o náročnosti výuky v jednotlivých měsících.

9.3 Do budoucna

Na základě zkušeností z období vývoje IS během uplynulých dvou let se domnívám, že informační technologie obecně se budou na základních školách dále rychle a úspěšně rozvíjet. Na začátku tvorby první části IS v rámci bakalářské práce byla práce s informačními technologiemi pro školy značně obtížná. Postupně se převážně díky úsilí zaměstnanců školy povedlo spustit jednak www stránky, informační systém školy, různé podpůrné vybavení pro administraci běžných úkonů a také pořídit spoustu IT vybavení pro jednotlivé třídy a jednotlivé učitele. Dle mého názoru se tak postupně z IT vybavení stávají naprosto standardní učební a pracovní pomůcky pro zaměstnance, žáky a rodiče. Zde mě nejvíce zaujal posun od původního zaměření na pouhé pořizování technického vybavení ke snaze o skutečné praktické použití. Z počátku se řešilo vybavení učeben dataprojektory, interaktivními tabulemi, pořízení počítačů pro učitele, vybavení učeben pro žáky počítači, pořízení serverů a potřebné infrastruktury, spuštění informačního systému, pořízení výukových programů, atd. Nyní se však školy zaměřují na efektivní využití těchto konkrétních technologií, což je, dle mého názoru, dobrá cesta, jak pokračovat.

Literatura

- [1] PIPREK, Tomáš. *IS pro Základní Školu U Kapličky 959, Orlová*. Ostrava, 2008. 40 s. Bakalářská práce. VŠB - Technická univerzita Ostrava
- [2] ZIMMERMAN, Radek. *IS pro Základní školu v Lošticích*. Ostrava, 2008. 43 s. Bakalářská práce. VŠB - Technická univerzita Ostrava
- [3] GRUDL, David. *NETTE Framework* [online]. 2011 [cit. 2011-03-28]. Dokumentace.
<<http://nette.org/cs/dokumentace>>.
- [4] *Wikipedia.cz* [online]. 2010 [cit. 2010-10-03]. Hypertext Transfer protokol. Dostupné z WWW: <http://cs.wikipedia.org/wiki/Hypertext_Transfer_Protocol>.
- [5] *Wikipedia.cz* [online]. 2010 [cit. 2011-03-28]. Asymetrická kryptografie.
<http://cs.wikipedia.org/wiki/Asymetrick%C3%A1_kryptografie>.
- [6] INTERNET CZ, a.s. *Forpsi.com* [online]. 2010 [cit. 2011-03-28]. Ceník.
<<https://www.forpsi.com/webhosting/pricelist.php>>.
- [7] ZONER software, a.s. *Czechia.com* [online]. 2010 [cit. 2011-03-28]. SSL certifikáty.
<<http://www.czechia.com/clanek/ssl-certifikaty>>.
- [8] *Dibi* [online]. 2010 [cit. 2011-03-28]. Dokumentace.
<<http://dibiphp.com/cs/>>.
- [9] *Wikipedia.cz* [online]. 2010 [cit. 2011-03-28]. Active Record.
<http://cs.wikipedia.org/wiki/Active_Record>.
- [10] *Google Apps* [online]. 2010 [cit. 2011-03-28]. Dokumentace.
<<http://www.google.com/apps>>.
- [11] ZONER software, a.s. *Czechia.com* [online]. 2010 [cit. 2011-03-29]. Certifikáty, Verisign.
<<http://www.czechia.com/clanek/ssl-verisign/>>.

- [12] *Wikipedia.cz* [online]. 2010 [cit. 2011-03-29]. Model-view-controller.
<<http://cs.wikipedia.org/wiki/Model-view-controller>>.
- [13] GRUDL, David. *NETTE Framework* [online]. 2010 [cit. 2011-03-29]. Licenční politika.
<<http://nette.org/cs/licence>>.
- [14] *Firebug* [online]. 2010 [cit. 2011-03-29]. Firebug.
<http://getfirebug.com/>
- [15] *Root.cz* [online]. 2010 [cit. 2011-03-29]. Velký test PHP frameworků.
<<http://www.root.cz/clanky/velky-test-php-frameworku-zend-nette-php-a-ror/>>.
- [16] *MPDF* [online]. 2010 [cit. 2011-03-29]. MPDF.
<<http://mpdf.bpm1.com/>>.
- [17] *Open Source Initiative* [online]. 2010 [cit. 2011-03-29]. GNU General Public License Versions.
<<http://www.opensource.org/licenses/gpl-license.php>>.
- [18] LACKO, Luboslav. *Datové sklady, OLAP a dolování dat*. vydání první. Brno : Computer Press, 2003. 486 s. ISBN 80-7226-969-0.
- [19] ŠARMANOVÁ, Jana. *Metody analýzy dat*. Ostrava : VŠB - Technická univerzita Ostrava, 2010. 57 s.
- [20] ŠARMANOVÁ, Jana. *TEORIE ZPRACOVÁNÍ DAT*. Ostrava : VŠB - Technická univerzita Ostrava, 2003. 74 s.
- [21] ŠARMANOVÁ, Jana. *INFORMAČNÍ SYSTÉMY A DATOVÉ SKLADY*. první. Ostrava : VŠB - Technická univerzita Ostrava, 2007. 169 s. ISBN 978-80-248-1500-8.

Obsah CD

/root	kořenový adresář CD
abstrakt-CZ.txt	abstrakt v českém jazyce
abstrakt-EN.txt	abstrakt v anglickém jazyce
/database/	adresář s sql soubory pro vytvoření struktury databáze
/instalace/	adresář s instalačními soubory pro Apache a MySQL
/text/	adresář s texty
/prilohy/	adresář s přílohami
pip009-dp-prace.pdf	text diplomové práce
/zdrojove_kody	
/IS/	zdrojové kódy pro IS
/3rdParty/	soubory třetích stran pro NETTE framework
/admin/	soubory pro uživatele s rolí administrátor
/app/	soubory modelů, presenterů a šablon pro NETTE framework
/dokument_root/	soubory s css styly a grafikou
/sekretarka/	soubory pro uživatele s rolí sekretářka
/analyza_dat	
/ PIP009 - Analysis Project/	soubory projektu Analysis services z Visual Studia 2008
/ PIP009 - Integration Project/	soubory projektu Integration services z Visual Studia 2008

Přílohy

A. Ukázky www prezentací jednotlivých škol

V textu diplomové práce nejsou vzhledem k jejímu zaměření uvedeny ukázky www prezentací škol. Jednotlivé ukázky konkrétních částí www stránek a ukázky statistik přístupů jsou uvedeny v této příloze.

/text/prilohy/A-ukazky-www-stranek.pdf

B. Programátorská dokumentace – Použití protokolu HTTPS

V této příloze je uveden postup pro konfiguraci www serveru Apache pro provoz protokolu HTTPS.

/text/prilohy/B-programatorska-dokumentace-pouziti-https.pdf

C. Programátorská dokumentace

Programátorská dokumentace vygenerovaná pomocí programu phpDocumentor.

/text/prilohy/C-programatorska_dokumentace/

D. Kompletní datová analýza

V diplomové práci je uvedena datová analýza pouze pro nové funkce IS. Kompletní datová analýza je uvedena v této příloze.

/text/prilohy/D-kompletni-datova-analyza.pdf

E. Ukázky výpisů známek

V této příloze jsou uvedeny ukázky PDF výpisů známek vygenerovaných v IS prostřednictvím funkce Výpisy známek v PDF, kterou má k dispozici učitel.

/text/prilohy/E-ukazky-pdf-vypisu-znamek.pdf

F. Ukázky existujících výpisů známek

V současnosti se používá pro pololetní výpisy známek program DM Vysvědčení. Výpisy vytvořené v tomto programu sloužily jako podklad pro implementaci výpisů v IS. Jejich ukázka je v této příloze.

/text/prilohy/F-ukazka-existujicich-vypisu-znamek.pdf

G. Uživatelská dokumentace

V této příloze je uvedena základní uživatelská dokumentace pro nově implementované funkce IS, které se týkají převážně učitele.

</text/prilohy/G-uzivatelska-dokumentace.pdf>